

Direction des bibliothèques

AVIS

Ce document a été numérisé par la Division de la gestion des documents et des archives de l'Université de Montréal.

L'auteur a autorisé l'Université de Montréal à reproduire et diffuser, en totalité ou en partie, par quelque moyen que ce soit et sur quelque support que ce soit, et exclusivement à des fins non lucratives d'enseignement et de recherche, des copies de ce mémoire ou de cette thèse.

L'auteur et les coauteurs le cas échéant conservent la propriété du droit d'auteur et des droits moraux qui protègent ce document. Ni la thèse ou le mémoire, ni des extraits substantiels de ce document, ne doivent être imprimés ou autrement reproduits sans l'autorisation de l'auteur.

Afin de se conformer à la Loi canadienne sur la protection des renseignements personnels, quelques formulaires secondaires, coordonnées ou signatures intégrées au texte ont pu être enlevés de ce document. Bien que cela ait pu affecter la pagination, il n'y a aucun contenu manquant.

NOTICE

This document was digitized by the Records Management & Archives Division of Université de Montréal.

The author of this thesis or dissertation has granted a nonexclusive license allowing Université de Montréal to reproduce and publish the document, in part or in whole, and in any format, solely for noncommercial educational and research purposes.

The author and co-authors if applicable retain copyright ownership and moral rights in this document. Neither the whole thesis or dissertation, nor substantial extracts from it, may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms, contact information or signatures may have been removed from the document. While this may affect the document page count, it does not represent any loss of content from the document.

Université de Montréal

Le rôle épistémique de certaines simulations informatiques fondamentales en théorie de l'évolution

par
Martin Ledoux

Département de philosophie
Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de M.A. en philosophie

Mars 2008

©, Martin Ledoux, 2008



Université de Montréal
Faculté des études supérieures

Ce mémoire intitulé :

Le rôle épistémique de certaines simulations informatiques
fondamentales en théorie de l'évolution

présenté par :
Martin Ledoux

a été évalué par un jury composé des personnes suivantes :

M. JEAN-PIERRE MARQUIS
président-rapporteur

M. Frédéric Bouchard
directeur de recherche

M. FRANÇOIS LEPAGE
membre du jury

Résumé et mots-clés

Résumé

Ce mémoire porte sur le rôle épistémique que joue une simulation informatique canonique en théorie de l'évolution. La première partie présente un examen de la conception que propose Paul Humphreys du rôle épistémique des méthodes informatiques en science en général. Pour ce faire, une analyse du concept de patron computationnel (« computational template ») est effectuée ; puis, ce concept est placé dans son cadre plus large de modèle computationnel (« computational model »). Ceci permet de circonscrire une conception de la simulation informatique en tant qu'instrument expérimental. La deuxième partie, en s'appuyant sur cette conception, explore le rôle épistémique de la simulation d'agents au coeur des travaux d'Axelrod sur l'évolution de la coopération. Il devient alors possible d'identifier certains points faibles de la conception de Humphreys et de souligner le caractère plus régulateur qu'explicatif de la simulation informatique en théorie de l'évolution.

Mots-clés

philosophie, épistémologie, évolution, coopération, simulation, informatique

Abstract and Keywords

Abstract

In this master's thesis, the epistemic role played by a canonical computer simulation in the theory of evolution is explored. The first part presents an analysis of Paul Humphreys' conception of the epistemic role of computational methods in science. This is achieved by analyzing the concept of computational template and placing it in the broader context of the computational model. A view of the computer simulation qua experimental instrument is then sketched out. The second part, using the results of the analysis of Humphreys' view, explores the epistemic role of the agent-based simulations used by Axelrod in his work on the evolution of cooperation. It is then possible to identify certain weaknesses in Humphreys' conception, as well as confirm that computer simulations in the theory of evolution play an intuitive rather than explicative role.

Keywords

philosophy, epistemology, evolution, cooperation, simulation, computer science

Table des matières

Identification du jury	i
Résumé et mots-clés	i
Abstract and Keywords	ii
Remerciements	vi
Introduction	1
Science et simulation	1
Plan de la discussion	2
Quelques mises au point	3
1 L'empirisme computationnel de Humphreys	5
1.1 Vers une nouvelle conception ?	5
1.1.1 Les mathématiques tractables, moteur du progrès	5
1.1.2 La sémantique ne suffit pas	6
1.1.3 L'interprétation non détachable	8
1.1.4 Des extensions similaires	9
1.1.5 Une nouvelle méthode scientifique	11
1.1.6 Un changement de conception	11
1.2 Patrons et modèles computationnels	12
1.2.1 Patrons théoriques et patrons computationnels	12
1.2.2 Caractéristiques des patrons	13
1.2.3 Processus de construction des patrons	13
1.2.4 Modèles computationnels	15
1.2.5 L'empirisme computationnel	15
1.3 Critique de l'empirisme computationnel	16
1.3.1 La sous-détermination chronique	16
1.3.2 Réalisme ou instrumentalisme ?	18
1.3.3 L'opacité épistémique	18
1.3.4 Un pis-aller temporaire	20

1.3.5	Hors-la-loi ?	21
1.4	Conséquences de l'empirisme computationnel	21
1.4.1	En pratique plutôt qu'en principe	22
1.4.2	Un réalisme sélectif	22
1.4.3	La réorganisation des sciences	23
1.5	La nature des simulations informatiques	23
1.5.1	Définition et statut de la simulation informatique	24
1.5.2	La simulation <i>qua</i> instrument expérimental	26
2	La simulation d'Axelrod	30
2.1	Présentation de l'expérience	30
2.1.1	Le dilemme du prisonnier	30
2.1.2	La simulation informatique	31
2.2	Dégagement de la structure de modèle computationnel	33
2.2.1	Patron computationnel	33
2.2.2	Suppositions de construction	35
2.2.3	Ensemble de correction	37
2.2.4	Patron ou patrons ?	39
2.2.5	Relation entre patrons apparentés	42
2.2.6	Le modèle du dilemme itéré du prisonnier	43
2.3	La question de la simulation	44
2.4	Le rôle épistémique des simulations	47
2.5	Conclusion	48
	Discussion et conclusion	50
	Bibliographie	57

Liste des tableaux

2.1	Le dilemme du prisonnier	31
2.2	Exemple du dilemme itéré du prisonnier	32

Je remercie mon directeur de recherche, M. Frédéric Bouchard, pour
ses conseils judicieux et ses commentaires toujours appréciés.

Je remercie également ma conjointe, Sylvie Boudreau, pour
son soutien indéfectible.

Introduction

Science et simulation

Durant les cinquante dernières années, les scientifiques se sont appropriés un nombre croissant de méthodes computationnelles. Peu de temps après la Deuxième guerre mondiale, le premier ordinateur numérique programmable, ENIAC, fut mis à contribution pour simuler – sur une seule dimension – une explosion nucléaire (Humphreys, 2007, p. 65). ENIAC, occupant une pièce de près de 68 mètres carrés et pesant 27 tonnes, pouvait effectuer 357 multiplications ou 35 divisions (de nombres de 10 chiffres) par seconde (Wilkes, 1956).

En 2005, des chercheurs du Los Alamos National Laboratory ont effectué une simulation informatique mettant en oeuvre 2,64 millions d'atomes en mouvement. La simulation fut effectuée sur un ensemble de 768 ordinateurs comptant chacun 8 192 processeurs¹.

Un tel progrès dans la capacité qu'ont les chercheurs scientifiques à effectuer des simulations par ordinateur a de quoi susciter l'admiration, l'incrédulité peut-être, mais surtout, l'interrogation. Quel impact l'accroissement de nos capacités computationnelles a-t-il eu sur nos capacités épistémiques ? Quel genre de connaissance scientifique peut nous apporter la simulation informatique ? Dans quelle mesure la technologie disponible rend possible – ou contraint – la recherche scientifique ? La simulation informatique ouvre-t-elle la porte sur un nouveau mode de recherche et de connaissance scientifique ?

La conception de l'empirisme computationnel que propose le philosophe des sciences Paul Humphreys (2004) constitue une réponse à ces questions. Mais la conception que Humphreys (2004) propose, si elle est potentiellement applicable à toutes les utilisations de simulations dans toutes les disciplines scientifiques, est surtout illustrée par des exemples tirés de la physique.

Or, plusieurs raisons motivent que l'on s'intéresse à l'application de cette conception à la biologie. D'une part, on constate l'utilisation grandissante des simulations informatiques en biologie² ; d'autre part, on doit admettre la spécificité des sciences biologiques, par exemple l'historicité inhérente à tout processus évolutionnaire ; enfin, il existe un risque que la simulation informatique, par son faible coût et sa simplicité logistique, remplace les méthodes empiriques traditionnelles et fasse de la biologie une science fondée *a priori* (Humphreys, 2004).

¹Pour plus de détails, voir http://www.lanl.gov/news/index.php/fuseaction/home.story/story_id/7428.

²Deux survols récents de cette utilisation sont (Kitano, 2002) et (Keller, 2005).

Ces trois raisons motivent sans peine notre tentative d'appliquer la conception de Humphreys (2004) à une simulation informatique canonique en théorie de l'évolution : la simulation d'agents, au coeur des travaux d'Axelrod (1984, 1997) sur l'évolution de la coopération en tant que stratégie évolutionnaire stable (Maynard Smith, 1982). Cette étude d'un cas particulier nous semble fondamentale pour comprendre comment les simulations informatiques pourraient fonctionner en biologie évolutionnaire en général.

Nous visons donc, par le présent mémoire, à répondre à cette question : **quel est le rôle épistémique de certaines simulations informatiques fondamentales en théorie de l'évolution ?** Pour atteindre cet objectif, nous devrons d'abord (au premier chapitre) analyser de manière critique la conception que propose Humphreys (2004). Ensuite, nous pourrons (au deuxième chapitre) illustrer cette conception par l'exemple des simulations d'Axelrod (1984, 1997).

Plan de la discussion

Concrètement, la discussion se déroulera en six moments. Les trois premiers moments nous permettront de répondre à notre premier sous-objectif, l'analyse critique de la position de Humphreys (2004) :

- En premier lieu (sections 1.1 et 1.2), nous présenterons la conception de l'empirisme computationnel de Humphreys (2004). Nous aborderons d'abord les cinq constats motivant cette conception. Puis, nous analyserons les deux concepts centraux à la conception : le patron computationnel et le modèle computationnel.
- En deuxième lieu (sections 1.3 et 1.4), nous aborderons certaines critiques adressées à la conception de Humphreys (2004). Nous aurons donc l'occasion de nous pencher sur les concepts de sous-détermination, d'instrumentalisme et d'opacité de modèle. Nous présenterons également certaines conséquences possibles de l'empirisme computationnel, ce qui nous permettra de considérer la variété particulière de réalisme à laquelle Humphreys (2004) adhère.

Nous aborderons également la question de la réorganisation des sciences selon le patron computationnel utilisé plutôt que selon l'objet d'étude. En d'autres termes, la méthode serait plus informative que l'objet en ce qui a trait à la caractérisation des disciplines scientifiques. Ainsi, la distinction rigide (p. ex. entre biologie évolutionnaire et sciences économiques) devient arbitraire lorsqu'on s'attarde aux méthodes partagées (ici, la théorie des jeux) maintenant déterminantes pour le déploiement de ces disciplines.

- En troisième lieu (section 1.5), nous traiterons de la place particulière des simulations informatiques dans la conception de Humphreys (2004). Nous analyserons donc la définition et le statut de la simulation informatique dans cette conception. Nous aurons aussi l'occasion de confronter cette dernière à une critique provenant de Hacking (1981).

Nous viserons, aux deux moments suivants, l'atteinte de notre second sous-objectif : l'illustration de la conception de Humphreys (2004) par l'exemple des simulations d'Axelrod (1984,

1997) :

- D’abord (section 2.1), nous présenterons de manière détaillée les différentes expériences effectuées par Axelrod dans le cadre de ses recherches sur l’évolution de la coopération (Axelrod, 1984, 1997).
- En cinquième lieu (section 2.2), nous dégagerons des expériences d’Axelrod une structure de modèle computationnel au sens de Humphreys (2004). Nous aurons donc à nous pencher sur la question des relations entre patrons computationnels apparentés, ainsi que sur le statut des expériences d’Axelrod *qua* simulation au sens de Humphreys (2004).

Enfin, nous présenterons nos conclusions générales, qui apporteront un élément de réponse à notre question de recherche et éclaireront au moins partiellement le rôle épistémique que jouent – ou peuvent jouer – certaines simulations informatiques fondamentales en théorie de l’évolution. En particulier, nous identifierons des aspects de la conception qui doivent être modifiés ; nous proposerons d’ailleurs quelques moyens possibles d’effectuer ces modifications. Ceci conclura le présent mémoire, mais certainement pas la discussion sur la question.

Quelques mises au point

Avant d’entamer ce parcours, nous devons effectuer quelques mises au point, voire mises en garde. D’une part, Humphreys (2004) justifie sa conception sur cinq constats, dont l’un d’eux (le premier, présenté à la section 1.1.1) repose sur la prémisse que la science progresse dans le temps. Or, cette prémisse n’est jamais justifiée ou même remise en question par Humphreys (2004). Il se contente de préciser que « *scientific progress involves a temporally ordered sequences of stages* » (Humphreys, 2004, p. 154), ce qui est soit un truisme (le progrès n’est-il pas *par définition* une suite chronologiquement ordonnée d’étapes ?), soit la défense d’une conception continuiste de la science.

Nous sommes conscients que la notion de progrès scientifique – particulièrement une conception continuiste de celui-ci – est loin de faire l’unanimité en philosophie des sciences et que la prémisse utilisée par Humphreys (2004) ne saurait résister à l’examen philosophique sévère d’un Kuhn (1996) ou d’un Feyerabend (1978). Nous demandons – en notre nom mais aussi au nom de Humphreys (2004) – l’indulgence du lecteur et son acceptation provisoire d’une conception naïve du progrès scientifique.

Une autre mise en garde concerne la diversité des exemples scientifiques utilisés dans le présent texte, diversité qui provient de trois sources. D’une part, Humphreys utilise à des fins rhétoriques des exemples tirées des sciences physiques, en particulier de la thermodynamique. D’autre part, l’un des objectifs principaux de Humphreys (2004) est de montrer la transdisciplinarité des méthodes computationnelles : cette démonstration s’accompagne donc d’un recours à des exemples tirés de la climatologie, de l’astronomie, de la biologie, etc. Enfin, l’objectif du présent texte est d’appliquer la conception de Humphreys (2004) à la simulation d’Axelrod (1984), simulation qui a joué un aussi grand rôle en biologie évolutionnaire qu’en sciences po-

litiques. Ces trois raisons expliquent et justifient sans peine la diversité des exemples présentés.

Une autre mise au point touche notre utilisation des qualificatifs « computationnel » et « informatique ». Nous utiliserons le premier lorsqu'il sera question d'un dispositif de calcul en général (p. ex. un ordinateur, mais aussi une calculatrice, une règle à calcul, un humain équipé de papier et d'un crayon, etc.) ; et le second lorsqu'il sera plus spécifiquement question d'une mise en oeuvre d'un ordinateur (p. ex. dans le cas d'une simulation informatique).

Enfin, l'exemple de la simulation d'Axelrod, présenté en deuxième partie, n'est que cela : un exemple. Nous n'utilisons pas l'expérience d'Axelrod afin de démontrer le bien-fondé de la conception de Humphreys, mais plutôt afin d'illustrer cette dernière. Cette illustration nous permettra également de considérer la conception de Humphreys non plus sous l'angle de la physique (domaine duquel Humphreys tire la plupart de ses exemples) mais plutôt de la biologie évolutionnaire.

Ces mises au point étant faites, penchons-nous maintenant sur cette conception : l'empirisme computationnel de Humphreys.

Chapitre 1

L'empirisme computationnel de Humphreys

Dans cette première partie, nous considérerons d'abord les circonstances justifiant une nouvelle conception de la méthode scientifique, l'empirisme computationnel de Humphreys (2004). Puis, nous en analyserons la teneur et procéderons à une critique de cette conception. Enfin, nous explorerons de manière critique un sous-domaine de la conception de Humphreys : la simulation informatique. Cela nous permettra, en deuxième partie, d'examiner notre cas d'étude : la simulation d'Axelrod (1984, 1997), visant à étudier les conditions de possibilité de l'émergence de la coopération entre agents.

1.1 Vers une nouvelle conception ?

Humphreys (1995a) présente cinq constats qu'il interprète comme justifiant l'adoption d'une nouvelle conception de la méthode scientifique.

1.1.1 Les mathématiques tractables, moteur du progrès

En premier lieu, Humphreys (1995a, 2004) affirme que le moteur principal du progrès scientifique¹ provient du développement de mathématiques tractables² ; un constat qui devrait aller de soi :

« This is such an obvious truth that it would not be worth stating had it not been pushed to the background in methodological studies by the emphasis on metamathematics and logic. » (Humphreys, 1995a, p. 120)

En effet, la forme logique, si elle permet une grande généralité dans le propos, masque toute question de tractabilité, car elle est détachée de toute forme mathématique particulière. L'accent

¹Tel que précisé dans l'introduction, c'est un concept naïf de progrès scientifique qui est utilisé ici.

²Un problème computationnel *intractable* peut être en principe résolu, mais requiert trop de temps ou d'espace mémoire pour l'être en pratique (Sipser, 1997, p. 305). Un problème tractable est donc un problème soluble non seulement en principe mais également en pratique.

mis sur la forme logique s'est donc fait au prix d'une mise entre parenthèses de toute question de tractabilité.

Malgré cette occultation, il demeure qu'historiquement, c'est le développement de méthodes mathématiques calculables en pratique qui aurait principalement alimenté le progrès scientifique :

« Whenever you have a sudden increase in usable mathematics, there will be a sudden, concomitant increase in scientific progress in the area affected. » (Humphreys, 2004, p. 55)

L'idée n'est pas nouvelle : Lakatos soulignait déjà que *« [t]he real difficulties for the theoretical scientist arise rather from the mathematical difficulties of the programme than from anomalies »* (Lakatos, 1970, p. 137).

Notons que d'autres auteurs (p. ex. Kline (1971)) ont défendu le raisonnement inverse : les exigences de la science alimenteraient le développement des mathématiques. D'une part, on peut se demander avec Humphreys dans quelle mesure ce lien est fort : *« it is easy to overstate that kind of influence »* (Humphreys, 2004, p. 55). D'autre part, même si l'influence en question était forte, la cause (proximale) des progrès scientifiques futurs serait quand même le développement des mathématiques, plutôt que les exigences théoriques ayant mené à celui-ci.

Donc, la tractabilité mathématique constitue une voie privilégiée du progrès scientifique. C'est donc dire que les formes mathématiques particulières, c'est-à-dire la syntaxe mathématique, sont cruciales : on ne peut se contenter de considérer les formes logiques ou abstraites, hors de toute considération computationnelle.

1.1.2 La sémantique ne suffit pas

Au vingtième siècle, deux conceptions différentes des théories scientifiques se sont mesurées l'une à l'autre. La CONCEPTION SYNTAXIQUE des théories scientifiques repose sur un fondement logique important :

« A scientific theory consists of two parts. One part is an abstract logical calculus. In addition to the vocabulary of logic, this calculus includes the primitive symbols of the theory, and the logical structure of the theory is fixed by stating the axioms or postulates of the theory in terms of its primitive symbols. » (Suppes, 1967, p. 56)

En d'autres termes, une théorie scientifique est constituée d'axiomes (postulats), desquels on peut déduire – par le calcul logique adopté – des théorèmes. Le modèle D-N (déductif-nomologique) de Hempel (2004, p. 77-83) est un exemple éclairant d'approche syntaxique : à partir d'un certain nombre de lois L_i et d'énoncés de faits C_j , on déduit *logiquement* une proposition explanandum E . En d'autres termes :

$$(L_1 \wedge L_2 \wedge \dots \wedge L_m \wedge C_1 \wedge C_2 \wedge \dots \wedge C_n) \Rightarrow E$$

Le modèle D-N illustre admirablement le rôle explicatif vital des lois scientifiques dans la conception syntaxique. Il éclaire également la fusion qui est réalisée entre la logique et l'empir-

risme : en effet, un certain nombre de prémisses (les C_j) de la déduction logique, ainsi que la conclusion (E) de cette dernière, sont des énoncés de faits empiriques.

Une telle approche syntaxique a le mérite d'être hautement schématisée et de bénéficier de la limpidité de la logique : en effet, ce formalisme garantit l'absence de toute univocité. Cependant, plusieurs problèmes entachent la conception syntaxique des théories scientifiques (van Frassen, 1972) :

- « *[T]he difficulties of implementing the axiomatic method for actual theories, both in principle and in practice, are now well known.* » (van Frassen, 1972, p. 305). Ce problème est particulièrement présent dans le cas de théories complexes (Suppes, 1967).
- De plus, une formulation axiomatique d'une théorie ne présente cette dernière que sous un seul jour, alors que d'autres formulations sont possibles – et peut-être plus fructueuses.
- Enfin, l'approche logique insiste sur une formulation purement syntaxique d'une théorie, alors qu'une formulation plus « libérale » (le terme est de van Fraassen) pourrait s'avérer plus utile.

La CONCEPTION SÉMANTIQUE des théories scientifiques vise à répondre à ces difficultés. Plutôt que de formuler une théorie scientifique en termes d'ensemble d'axiomes muni d'un calcul logique, l'approche sémantique identifie une théorie à l'ensemble de ses modèles :

« *From this point of view, the essential job of a scientific theory is to provide us with a family of models, to be used for the representation of empirical phenomena.* » (van Frassen, 1972, p. 310)

Notons que le terme de modèle prend un sens différent dans les conceptions syntaxique et sémantique :

« *Interpretations that make all the statements in the set true—logicians call these "models"—may be given for certain theories. In our discussion, a model is not such an interpretation, matching statements to a set of objects that bear certain relations among themselves, but the set of objects itself.* » (Lloyd, 1994, p. 15)

La clé n'est plus, comme dans la conception syntaxique, la déduction logique mais plutôt l'ISOMORPHISME entre modèles théoriques (qui forment *in fine* la théorie elle-même) et modèles empiriques (les phénomènes dont on veut rendre compte ; l'*explanandum*). En particulier, on peut caractériser une théorie en définissant la classe des modèles qui doit lui correspondre (Lloyd, 1994, p. 15). Cette caractérisation sémantique, contrairement à une caractérisation axiomatique, peut être plus ou moins formelle.

Et pour Humphreys (2004), c'est sur ce point que le bât blesse. En effet, à la lumière du premier constat, on doit conclure que la conception sémantique des théories (van Frassen, 1972; Suppes, 1967, 2002) est inadéquate, pour au moins deux raisons. D'une part, elle ne permet pas de rendre compte de l'application des théories scientifiques aux systèmes réels ; et d'autre part, elle ne permet pas de rendre compte du progrès scientifique. Examinons ces deux raisons de plus près.

On sait que la conception sémantique dévalue la forme syntaxique particulière afin de dégager une structure sémantique générale. Or, à une même structure sémantique peuvent corres-

pondre plusieurs formes syntaxiques inégales en tractabilité :

« The change from rectilinear to polar coordinates, from a laboratory frame to a center-of-mass frame, from a Lagrangian to a Hamiltonian representation [...] all these and more make the difference between a computationally intractable and a computationally tractable representation. » » (Humphreys, 2004, p. 97)

Pour qu'une théorie scientifique puisse être appliquée à un système réel, il faut faire intervenir de telles représentations – et même plus, des représentations *tractables* ! En passant sous silence l'importance de celles-ci, la conception sémantique des théories ignore du même coup l'applicabilité des théories aux systèmes réels.

D'autre part, parce que le progrès scientifique dépend du développement de mathématiques tractables, elles-mêmes ignorées par la conception sémantique, cette dernière est donc inadéquate pour rendre compte du progrès en science.

Ainsi, pour ces deux raisons, on ne peut détacher la forme syntaxique de la théorie. Une autre composante d'une théorie est, selon Humphreys, tout aussi constitutive : l'interprétation.

1.1.3 L'interprétation non détachable

En effet, selon Humphreys, les théories formulées mathématiquement sont en général accompagnées d'une interprétation, qui en justifie l'utilisation. Aucune conception – syntaxique ou sémantique – des théories scientifiques ne peut se permettre de dissocier le formalisme de l'interprétation, pour au moins trois raisons.

En premier lieu, en détachant l'interprétation du formalisme, on court le risque de s'exposer au problème de la sous-détermination. En effet, à partir du moment où on rejette l'interprétation originale,

« the same formal theory can be given a variety of distinct interpretations, all of which are compatible with the empirical data and the formal constraints imposed by the theory, but each of these interpretations employs a different ontology. »
(Humphreys, 1995a, p. 123)

Or, le problème de la sous-détermination, parce qu'il permet la réinterprétation *a volo* d'une théorie, en utilisant une ontologie différente, constitue un réel danger pour quiconque veut défendre une position réaliste³.

En deuxième lieu, l'interprétation jointe initialement à une théorie en constitue une justification dont on ne saurait se priver. Humphreys (1995b, 2004), à cet égard, donne en exemple trois dérivations différentes de l'équation de la chaleur. Or, dans la première de celles-ci,

« the interpretation of the function $u(x, t)$ represented the temperature gradient in a perfectly insulated cylindrical conductor, and this interpretation is central to the

³Dont Humphreys. En effet, dans une note de bas de page (Humphreys, 2000, p. 30), il signale que l'expression « empiriste antiréaliste » n'est pas un pléonisme et conclut : « *I, for one, am an empirical realist* ». Quatre ans plus tard, la même note réapparaît presque textuellement (Humphreys, 2004, p. 17), mais cette fois-ci amputée de cet aveu de réalisme – ce qu'il est plausible d'expliquer par le fait que Humphreys, dans ce dernier ouvrage, défend plutôt une position de « réalisme sélectif ».

decision whether or not the diffusion equation is a correct representation of the flow of heat in a given metal bar. » (Humphreys, 2004, p. 80)

Ainsi, en détachant l'interprétation privilégiée et en ne conservant que le formalisme, on perd du même coup la justification initiale de ce dernier. De même, des équations différentielles en physique sont fondées – et sont accompagnées – de principes fondamentaux, de suppositions ontologiques, d'approximations, d'idéalisations, etc. Tous ces éléments ne peuvent être justifiés que si on les considère à la lumière de l'interprétation jointe à la théorie (Humphreys, 1995a, p. 124).

En troisième lieu, en l'absence d'interprétation privilégiée d'une théorie, comment ajuster celle-ci si elle s'avère réfutée par des données empiriques ? L'interprétation pourrait nous indiquer que ces déviations étaient à prévoir, tout en nous montrant comment ajuster la théorie. Par contre, une théorie sans interprétation, dès qu'elle est réfutée par les faits, semble devoir être éliminée sans autre forme de procès. L'interprétation privilégiée d'une théorie constitue donc en quelque sorte une protection contre le réfutationnisme de Popper, en nous permettant de corriger et de conserver une théorie scientifique, tout en préservant sa scientificité :

« [J]e n'exigerai pas d'un système scientifique qu'il puisse être choisi, une fois pour toutes, dans une acception positive mais j'exigerai que sa forme logique soit telle qu'il puisse être distingué, au moyen de tests empiriques, dans une acception négative : un système faisant partie de la science empirique doit pouvoir être réfuté par l'expérience. » (Popper, 1978, p. 37)

Il y a donc au moins trois raisons de conserver l'interprétation privilégiée jointe à une théorie scientifique. Ainsi, toute conception des théories scientifiques devra préserver le caractère non détachable de l'interprétation.

1.1.4 Des extensions similaires

L'avant-dernier constat concerne l'extension de nos moyens computationnels. Humphreys estime que l'extension de nos habiletés mathématiques naturelles par des dispositifs computationnels (ordinateurs, calculatrices, règles à calcul, humain équipé de papier et d'un crayon, etc.) est analogue à l'extension de nos habiletés perceptuelles naturelles par des instruments scientifiques (Humphreys, 1995a, p. 126 ; 2000, p. 21 ; 2004, p. 5). Ainsi, selon Humphreys, de la même manière que nous étendons notre vision par l'utilisation d'un microscope ou d'un télescope, nous étendons nos capacités mathématiques par l'utilisation d'ordinateurs ou autres dispositifs computationnels.

À cet égard, Humphreys (2004, p. 116-21) compare la technique de simulation informatique au microscope à puissance variable.

En premier lieu, tant le microscope que la simulation sont calibrés en les appliquant à des objets connus par d'autres moyens et en vérifiant que les résultats obtenus sont conformes à la structure observée. Par exemple, on vérifie qu'une grille très fine, observée au microscope, conserve sa structure. De même, on vérifie que la simulation produit les mêmes solutions que

des méthodes analytiques, dans les cas où ces méthodes sont tractables. On utilise ainsi dans les deux cas l'ARGUMENT DE CHEVAUCHEMENT (*overlap argument*) : puisque le domaine de l'instrument et le domaine déjà observé se chevauchent, on peut calibrer l'instrument.

En second lieu, Humphreys note qu'il est préférable – voire nécessaire – de connaître le fonctionnement de l'instrument, microscope ou simulation, afin de pouvoir l'utiliser convenablement. Par exemple, dans le cas du microscope, la connaissance du fonctionnement de cet instrument permet d'éviter d'interpréter incorrectement des artefacts optiques ; alors que dans le cas de la simulation, on doit être à l'affût des erreurs d'arrondi⁴. Il est clair que Humphreys récuse la position de Hacking, laquelle se résume par la formule « *[o]ne needs theory to make a microscope. You do not need theory to use one* » (Hacking, 1981, p. 309). Il semble au contraire qu'une connaissance du fonctionnement de l'instrument (empirique ou mathématique) soit nécessaire pour éviter de tomber dans l'erreur (nous reviendrons à cet argument à la section 1.5.2).

En dernier lieu, tant dans le cas du microscope que de la simulation, c'est le POUVOIR SÉPARATEUR (*resolving power*) qui constitue une contrainte à notre compréhension. Une image au microscope ne sera jamais plus détaillée que ce que la résolution optique de l'instrument permet. De même, une simulation aura une résolution finie, correspondant par exemple à sa granularité temporelle ou spatiale, au-delà de laquelle tout phénomène restera caché. Par exemple, les simulations météorologiques utilisant une granularité de l'ordre du mètre ne peuvent rendre compte des phénomènes plus petits :

« *[m]ismatches between the scale of these processes and computationally realizable grid scales in global models is a well-known problem of Earth system science* » (Harvey et al., 1997, p. 15)

En d'autres termes, « *[t]he degree of detail available is often limited by technological constraints such as the speed of the computational device and its memory capacity* » (Humphreys, 1995a, p. 126).

Ces trois raisons justifient donc l'emploi de la métaphore du microscope à puissance variable. Comme toute métaphore, celle-ci demeure imparfaite. Cependant, elle permet d'éclairer deux aspects importants. D'une part, les arguments en faveur de l'utilisation des instruments scientifiques risquent de s'avérer utiles pour justifier l'utilisation de méthodes computationnelles. D'autre part, on confirme le rôle important des avancées technologiques (tant au niveau des instruments que de l'aspect computationnel des mathématiques, c'est-à-dire de la tractabilité de celles-ci) dans le progrès scientifique. Par exemple, une percée technologique pourrait mener à une résolution plus fine qui elle-même permettrait la compréhension plus grande d'un plus grand nombre de phénomènes.

⁴Humphreys (2004, p. 118) donne à cet égard un exemple éclairant : le logiciel MATLAB version 12, bien connu des physiciens et mathématiciens, évalue $[1 - (1 - 10^{-16})]$ à $1,1102 \cdot 10^{-16}$, mais $[1 - (1 - 10^{-17})]$ à 0 (zéro). Quiconque choisit d'ignorer le fonctionnement de ce logiciel s'expose à de désagréables surprises...

1.1.5 Une nouvelle méthode scientifique

Dernier constat, mais non le moindre : certaines techniques computationnelles constituent une nouvelle méthode scientifique, intermédiaire entre l'expérience physique et le calcul numérique.

Considérons à titre d'exemple les méthodes de Monte Carlo, et plus précisément l'algorithme de Metropolis-Hastings. Cet algorithme, afin d'approcher une distribution de probabilité, simule par ordinateur un ensemble de promenades aléatoires dans un espace mathématique. Ainsi, la solution obtenue par un tel algorithme ne provient ni d'une expérience empirique traditionnelle (le véritable système physique n'est pas utilisé), ni d'un simple calcul numérique (le modèle stochastique utilisé est fondé sur un modèle physique spécifique du système), ni d'une méthode mathématique analytique (puisque'il s'agit d'une approximation numérique). Il s'agirait donc d'une nouvelle méthode scientifique (Humphreys, 1995a, p. 125).

Une caractéristique intéressante de cet exemple est la nécessité de recourir à des méthodes probabilistes : « *Monte Carlo methods need to be employed to calculate them* » (Humphreys, 1995a, p. 125; nous soulignons). On retrouve donc l'idée que la complexité du phénomène requiert qu'on fasse appel à un nouveau mode de connaissance :

« *[T]his approach opens up a new mode of scientific understanding. I even consider that it is obliged to do so, because theory-based insight, the common road to understanding, is blocked by what I have called a complexity barrier [...]* » (Lenhard, 2006b, p. 609)

Déjà, Buffon (en 1777 ; voir à cet égard (Bentz, 1980)) avait proposé de calculer la valeur de π par un processus stochastique (Humphreys, 1994) mettant en oeuvre le lancer répété d'une aiguille de longueur L (où $L \leq 1$) sur un ensemble de lignes parallèles situées à 1 de distance l'une de l'autre. Puisque la probabilité que l'aiguille croise une des lignes est égale à $\frac{2L}{\pi}$, on peut ainsi estimer la valeur de la célèbre constante. Évidemment, il s'agit d'une méthode encore empirique (puisque'on doit physiquement lancer une aiguille), mais elle préfigure déjà les méthodes de Monte Carlo.

1.1.6 Un changement de conception

Les cinq constats présentés constituent un plaidoyer en faveur d'une nouvelle conception de la méthode scientifique. Cette nouvelle approche devra rendre compte du rôle des mathématiques tractables, de l'importance de la syntaxe et de l'interprétation non détachable, des similitudes avec l'extension de notre appareillage perceptuel et enfin, du caractère novateur de certaines techniques.

L'empirisme computationnel de Humphreys constitue un candidat plausible d'une telle conception. Employons-nous maintenant à l'analyser afin d'en évaluer l'adéquation.

1.2 Patrons et modèles computationnels

À la section précédente, nous avons considéré cinq constats qui semblent montrer le besoin d'une nouvelle conception de la méthode scientifique. Il est possible de préciser davantage ce besoin, en terme d'UNITÉ D'ANALYSE, concept comprenant pour Humphreys (2004, p. 57-9) les théories, lois scientifiques, programmes de recherche (Lakatos, 1970), paradigmes (Kuhn, 1996), modèles, etc. En d'autres termes, l'unité d'analyse est l'EXPLANANS, c'est-à-dire l'explication fournie pour un phénomène donnée.

Or, selon Humphreys (2004), il manque à notre compréhension de la méthode scientifique une nouvelle unité d'analyse, faisant justice à la place des mathématiques tractables en science ; une place qui ne peut être circonscrite par l'une ou l'autre des unités d'analyse énumérées ci-dessus. Humphreys (2004) propose les patrons computationnels comme nouvelle unité d'analyse rendant compte de cette réalité. Voyons en quoi ils consistent.

1.2.1 Patrons théoriques et patrons computationnels

Considérons l'un des principes centraux à la théorie de la mécanique classique : $F = m \frac{d^2y}{dt^2}$. Pour Humphreys, une telle équation différentielle n'est pas une loi⁵ au sens propre du terme :

« *The first thing to note about Newton's Second 'Law' is that it is only a schema, what I shall call a theoretical template. It describes a very general constraint on the relationship between any force, mass, and acceleration, but to use it in any given case, we need to specify a particular force function* » (Humphreys, 2004, p. 60)

Un PATRON THÉORIQUE (*theoretical template*) est donc un schéma mathématique général, exprimant déjà une relation sans pour autant être applicable tel quel à un système réel. Lorsqu'on transforme un patron théorique afin de le rendre applicable et tractable, on obtient un PATRON COMPUTATIONNEL (*computational template*). Ainsi, dans l'exemple qui nous intéresse, si on utilise la force gravitationnelle (donc $F = \frac{GMm}{r^2}$), on obtient l'équation $\frac{GMm}{r^2} = m \frac{d^2y}{dt^2}$, qui (avec les conditions initiales $y(t_0) = y_0$ et $\left[\frac{dy}{dt}\right]_{t_0} = 0$) constitue un patron computationnel qui peut être aisément résolu afin d'obtenir une solution générale (Humphreys, 2004, p. 61-2).

Bien sûr, certains patrons computationnels sont plus appliqués que d'autres. Les plus importants, selon Humphreys (2004, p. 61), sont les patrons computationnels à la limite du théorique : des patrons qui ne pourraient être plus abstraits sans devenir inapplicables. Tous les patrons computationnels, cependant, doivent partager cette propriété de pouvoir être appliqués à des systèmes réels⁶.

⁵Nous n'aborderons pas ici la question de la possibilité même de parler de lois en physique ; le lecteur intéressé pourra consulter avec profit les travaux de Cartwright (1983, 1999) à cet égard. Nous aurons également l'occasion de revenir sur la question à la section 1.3.5.

⁶Une autre propriété commune aux patrons computationnels serait leur similitude au « *general argument pattern* » de Kitcher (1989) : le squelette d'une explication accompagné de directives pour en faire une explication à part entière. Il existe cependant plusieurs différences entre le « *general argument pattern* » et le patron computationnel (Humphreys, 2004, p. 71-2).

1.2.2 Caractéristiques des patrons

Reprenons l'exemple du patron de la seconde loi de Newton. On note qu'il aurait été possible de choisir une autre force que la force gravitationnelle, par exemple électrostatique ou magnétique. D'une part, cela explique en quoi le patron théorique reste théorique : il ne peut être tel quel appliqué à un système réel. D'autre part, un patron théorique est de fait très versatile, puisqu'il peut être utilisé pour former des patrons computationnels très différents. Cette tension entre sous-détermination et versatilité constitue une caractéristique importante des patrons, qui sera explorée plus en détail à la section 1.2.3.

La mise entre guillemets du mot « law » par Humphreys révèle une seconde caractéristique : les patrons ne sont pas nécessairement fondés sur des lois ou théories. Humphreys (2004, p. 88) donne l'exemple du patron du processus de Poisson, basé sur quatre suppositions assez simples, par exemple la première :

« *[d]uring a small interval of time, the probability of observing one event in that interval is proportional to the length of that interval* » (Humphreys, 2004, p. 88)

Un tel patron est extrêmement versatile et est utilisé pour représenter des phénomènes aussi variés que le nombre de coquilles par page, le nombre de faux numéros téléphoniques composés en un jour, etc. (Ross, 1997, p. 145). Cependant, il apparaît clairement de cet exemple que « *templates can be constructed in the absence of anything that looks like a systematic theory about a specific domain of objects* » (Humphreys, 2004, p. 88) : dans tous les exemples d'application, les quatre suppositions du patron du processus de Poisson ne sont justifiées que sur des bases empiriques. Ainsi, « *[t]heories need play no role in constructing templates* » (Humphreys, 2004, p. 89).

Le fait qu'un patron puisse être dénué de toute dimension nomologique constitue une différence notable entre la conception de Humphreys (2004) et les approches syntaxiques traditionnelles (présentées à la section 1.1.2). Ces dernières donnent une grande importance aux lois, qui peuvent cependant être absentes d'un patron computationnel.

1.2.3 Processus de construction des patrons

À des fins pédagogiques, les patrons computationnels sont fréquemment présentés comme étant des « boîtes noires ». Cependant,

« *[a]side from a few cases, all computational templates are constructed, even though at the textbook level, or when an application has become routine, this construction may be concealed.* » (Humphreys, 2002, p. S5)

Le processus de construction d'un patron peut *grosso modo* se diviser en deux parties (Humphreys, 2004) : les SUPPOSITIONS DE CONSTRUCTION (*construction assumptions*) et l'ENSEMBLE DE CORRECTION (*correction set*). Examinons ces deux moments.

Les suppositions de construction fondent le patron computationnel et comprennent « *a combination of basic principles, idealizations, approximations, and analogical reasoning* » (Humphreys, 2004, p. 72). Un exemple que Humphreys utilise à plusieurs reprises (Humphreys,

1995b, p. 504-5 ; 2004, p. 73-4) est l'équation de diffusion de la chaleur. De la dérivation du patron computationnel $\frac{\partial u}{\partial t} = \frac{k}{s\rho} \cdot \frac{\partial^2 u}{\partial x^2}$ (où $u(x, t)$ est la fonction de température, s est la chaleur spécifique et ρ est la densité de la tige cylindrique), Humphreys dégage les différentes composantes jouant un rôle dans la construction :

- Des principes de base (p. ex. le principe de conservation).
- Des idéalizations (p. ex. le cylindre est un conducteur parfait).
- Des raisonnements par analogie (la chaleur *qua* entité fluide).

D'autres cas mettent également en jeu des abstractions⁷ ainsi que des approximations. Toutes ces composantes peuvent être regroupées selon un certain formalisme que propose Humphreys (2004, p. 78) : les suppositions formulées et utilisées lors de la construction d'un patron (théorique ou computationnel) forment un 5-tuple

suppositions = (ontologie, idéalizations, abstractions, contraintes, approximations)

Quelques remarques s'imposent. D'une part, les contraintes comprennent des lois ainsi que certains principes de base. D'autre part, l'ontologie comprend les entités représentées dans le patron, ainsi que les relations qui existent entre elles. Enfin, on note que l'ontologie est la seule composante des suppositions de construction qui ne soit pas purement formelle (nous aurons l'occasion de revenir sur cette remarque à la section 1.2.4, p. 15).

La seconde partie du processus de construction, qui n'est cependant pas immédiatement utilisée lors de ce processus, est l'ensemble de correction :

« *The principal purpose of the correction set is to indicate, in advance, the ways in which the computational template will need to be adjusted when (not if) it fails to match the empirical data.* » (Humphreys, 2004, p. 78)

L'ensemble de correction est moins structuré que les suppositions de construction et peut comporter certains des éléments suivants (Humphreys, 2002, p. S6 ; 2004, p. 78-9) :

- Relaxation d'idéalizations, p. ex. ne plus considérer une sphère comme étant parfaite.
- Relaxation d'abstractions, p. ex. introduire la notion de friction.
- Relaxation des contraintes, p. ex. considérer que la chaleur peut quitter le cylindre.
- Raffinement des approximations, p. ex. utiliser une valeur plus précise pour une constante ou encore utiliser une grille plus fine – augmenter la résolution – dans une simulation.
- Changement dans l'ontologie. Il s'agit d'un changement majeur ; « *[it] come[s] very close to starting a new construction process* » (Humphreys, 2004, p. 79).

Les suppositions de construction et l'ensemble de correction constituent donc les deux volets de la construction d'un patron computationnel, avec chacun un objectif différent : « *[u]nlike the computational template, where solvability is primary, approach to the truth is the aim of the correction set.* » (Humphreys, 2002, p. S6)

⁷Humphreys (2004) admet cinq sens au mot « abstraction », mais qui finalement se rejoignent (à peu près) tous dans la définition « *[abstraction] is the process of omitting properties or their related predicates from a system or a model, the omission process being either cognitive, representational, or physical, leaving the remaining properties or predicates unchanged.* » (Humphreys, 2004, p. 143).

Dans la conception de Humphreys, le patron, les suppositions de construction ainsi que l'ensemble de correction font partie d'une structure plus large : le modèle computationnel.

1.2.4 Modèles computationnels

Un modèle computationnel, formellement (Humphreys, 2002, p. S10), est un 6-tuple :

modèle computationnel = (patron computationnel, suppositions de construction, ensemble de correction, interprétation, justification initiale, représentation de sortie)

Ce 6-tuple constitue « *an autonomous object of study* » (Humphreys, 2004, p. 103), « *a broad and increasingly important type of applied unit [of analysis]* » (Humphreys, 2002, p. S2). Ainsi, dans un certain sens, un modèle computationnel réalise un patron théorique, au même titre qu'un modèle traditionnel réalise une théorie.

Les trois premiers éléments du 6-tuple nous étant déjà familiers, attardons-nous sur les derniers éléments.

L'interprétation constituait le sujet du troisième constat (section 1.1.3, p. 8) ; les mêmes remarques s'appliquent ici. D'une part, l'interprétation est non détachable (d'où son enchâssement dans le 6-tuple composant le modèle). D'autre part, elle joue le triple rôle de prévenir la sous-détermination, de fournir une justification initiale au modèle et de permettre la correction – plutôt que le rejet pur et simple – du modèle face à des données empiriques réfutant ce dernier.

Quant à la justification initiale, elle émane de l'influence combinée de l'interprétation et du processus de construction :

« The construction process provides a prior justification for the model even before it is tested against data. Rather than the model being merely hypothetical, there are ordinarily reasons, albeit reasons modulated by the correction set, for adopting the various assumptions that go into the construction of the model. » (Humphreys, 2002, p. S6)

Enfin, la représentation de sortie consiste simplement en un moyen d'obtenir une représentation humainement intelligible des résultats ou solutions du patron computationnel (p. ex. un affichage sous forme d'images en mouvement). Il s'agit d'un élément certes simple, mais très important lorsqu'on considère que certains patrons computationnels ont un fonctionnement interne échappant – de par la quantité de données utilisée, par exemple – à l'inspection humaine directe.

1.2.5 L'empirisme computationnel

Les différentes composantes vues jusqu'à maintenant – patrons théoriques et computationnels, modèles computationnels, simulations informatiques – sont rassemblées par Humphreys (2004) dans ce qu'il appelle la SCIENCE COMPUTATIONNELLE (*computational science*) et qu'il définit ainsi :

« [It] consists in the development, exploration, and implementation of computational models [...] of nonmathematical systems using concrete computational devices. » (Humphreys, 2004, p. 105-6)

Une autre expression utilisée antérieurement par Humphreys (1995a) nous apparaît encore plus éclairante quant à son projet : l'EMPIRISME COMPUTATIONNEL (*computational empiricism*). L'expression nous apparaît révélatrice pour trois raisons.

En premier lieu, elle semble référer au courant de l'empirisme logique. Ce n'est pas, nous dit Humphreys, une coïncidence :

« [i]t was chosen as a deliberate parallel to the term 'logical empiricism' so as to emphasize the fact that the logically oriented methods philosophers have used for the greater part of this century need to be replaced, at least for some purposes, by methods that are computationally oriented. » (Humphreys, 1995a, p. 119-20)

On doit donc prendre l'empirisme computationnel, non pas comme un adjuvant à une conception existante de la méthode scientifique (p. ex. l'empirisme « classique »), mais plutôt comme un compétiteur direct, dans certains domaines.

En second lieu, ce même lien avec l'empirisme logique nous laisse croire que la forme syntaxique (plutôt que la structure sémantique) redevient importante comme sujet d'étude – mais une forme syntaxique mathématique, computationnelle, et non simplement logique⁸.

En troisième lieu, l'empirisme computationnel est un empirisme – l'accent sur les dispositifs de calcul réalisés (p. ex. les ordinateurs) est donc présent d'emblée. Les contraintes physiques auxquelles sont soumis ces dispositifs (p. ex. vitesse de processeur, taille de l'espace mémoire) constituent des limites bien réelles à ce que la science peut accomplir.

1.3 Critique de l'empirisme computationnel

1.3.1 La sous-détermination chronique

Il y a lieu de s'interroger sur la sous-détermination des patrons et modèles computationnels, c'est-à-dire le fait qu'à un patron (ou modèle) donné puissent correspondre deux (ou plus de deux) systèmes réels différents. Cette sous-détermination peut provenir de trois sources distinctes mais interreliées.

En premier lieu, la versatilité des patrons et modèles, ainsi que la facilité avec laquelle on peut les corriger et modifier, pourrait engendrer une sous-détermination. Mais Humphreys se fait rassurant :

⁸À cet égard, soulignons brièvement un problème soulevé par ce caractère computationnel : peut-on encore parler d'approche syntaxique, si la syntaxe en question n'est pas logique mais plutôt mathématique, voire computationnelle ? Si on accepte la thèse de Church-Turing (indémontrée mais généralement acceptée), qui *grosso modo* établit une équivalence entre ce qui peut être calculé par un algorithme et ce qui peut être calculé par une machine de Turing, le problème se dissout : un patron exprimé sous une forme mathématique peut, par cette thèse, être exprimé sous une forme computationnelle et donc, sous une forme logique. Pour plus de détails sur cette thèse, consulter (Sipser, 1997, sect. 3.3).

« *To adopt the detachable interpretation view is to deeply misrepresent the epistemic status that templates have in scientific work. [...] To peel off the intended interpretation is to remove the justification we have for adopting the equations.* » (Humphreys, 2004, p. 80)

Ainsi, un patron computationnel, parce qu'il est muni d'une interprétation non détachable, est *suffisamment* déterminé par celle-ci – suffisamment du moins pour ne pas souffrir de problèmes de sous-détermination liés à la versatilité du patron !

En deuxième lieu, nous avons vu lors de l'examen du troisième constat (p. 8) que l'absence – ou possiblement la faiblesse – de l'interprétation peut mener à la sous-détermination. Nous pouvons ajouter à cette source l'absence d'arrimage à une théorie : le patron computationnel du processus de Poisson (p. 13) est manifestement sous-déterminé (il s'agit même, nous l'avons vu, d'une de ses forces).

En dernier lieu, les simulations informatiques manifestent une autre forme de sous-détermination, plus fondamentale. Dans une simulation informatique, toutes les entités et relations sont représentées par des nombres (dans la mémoire de l'ordinateur, par exemple). Chaque simulation détermine donc deux systèmes distincts : le « vrai » système réel faisant l'objet d'une simulation ; mais également, un ensemble de nombres régis par des relations particulières. Humphreys parle à cet égard d'un retour du pythagorisme :

« *Pythagoras and his followers believed that the world was made up of numbers, and contemporary anti-realists have relentlessly pointed out that any consistent theory can be reinterpreted to be 'about' natural numbers. [...] we can see that computer simulations represent neo-Pythagoreanism in a particularly pure form.* » (Humphreys, 2004, p. 140)

Ces trois sources différentes de sous-détermination pourraient constituer un problème grave, voire fatal, pour la conception que défend Humphreys. Il n'en est rien. Au contraire, celui-ci considère que la sous-détermination est en fait une aubaine :

« *The practical advantages of this versatility of equational forms is enormous – just imagine what doing science would be like if each physically distinct phenomenon had a completely different kind of mathematical representation!* » (Humphreys, 1995b, p. 511)

En effet, puisque ce sont les mathématiques tractables qui constituent une contrainte au progrès scientifique, on doit se réjouir du fait que les mêmes patrons computationnels tractables peuvent servir pour comprendre plusieurs phénomènes distincts – ou, en d'autres termes, que « *[t]he same equations have the same solutions* » (Feynman et al., 1965).

Ainsi, la sous-détermination ne constitue un problème que pour un réaliste ne s'intéressant qu'à ce qu'on peut connaître en principe. Si on accepte que la science est limitée par ce qui peut se faire en pratique – c'est-à-dire limitée par la tractabilité – alors la sous-détermination est un avantage : « *[t]he realist's nightmare is the instrumentalist's dream come true* » (Humphreys, 1995b, p. 512).

1.3.2 Réalisme ou instrumentalisme ?

Cette dernière citation soulève immédiatement une question : Humphreys est-il un RÉALISTE, qui admet une certaine correspondance entre les entités et relations du discours scientifique et les entités et relations existant réellement dans le monde ? Ou est-il plutôt un INSTRUMENTALISTE, qui nie cette correspondance mais reconnaît à la science un rôle instrumental ?

La question est d'autant plus importante qu'elle reflète un certain flou dans la position de Humphreys au fil des années. Dans certains écrits (p. ex. (Humphreys, 1995b, p. 512)), Humphreys semble s'afficher comme un défenseur de l'instrumentalisme. Dans d'autres écrits, il proclame sans gêne son adhésion au réalisme : « *I, for one, am an empirical realist.* » (Humphreys, 2000, p. 30)

L'incohérence, cependant, n'est qu'apparente, sitôt que l'on réalise que Humphreys se réclame d'une position intermédiaire, voire d'un compromis :

« *Realism is often taken to be a position about theories taken in abstraction from our attitude toward them. [...] But realists are rarely so unselective about, or so distanced from, the contents of their theories. Realism is ordinarily an attitude held by an individual or by a group toward parts of a theory, and the realist commitments are not usually read straight off the theory itself.* » (Humphreys, 2004, p. 83)

Il est possible de tirer de ce passage au moins deux conséquences. D'une part, le réalisme ne s'applique (ordinairement) qu'à des *parties* d'une théorie, et non à la théorie dans son ensemble. Ainsi, on peut s'attendre à ce que les parties d'une théorie qui ne font *pas* l'objet d'une attitude réaliste, ne jouent *in fine* qu'un rôle instrumental.

D'autre part, le réalisme n'est pas, selon Humphreys (2004), une propriété objective d'une théorie mais plutôt une attitude qu'a le théoricien face à sa création. Il y a donc une part interprétative dans le réalisme, qui devient plus un processus qu'un état : une herméneutique réaliste.

Ces deux conséquences, prises ensemble, nous entraînent vers un compromis qu'est le réalisme sélectif, la « solution évidente » :

« *The obvious solution is to be explicit about which parts of the theory are to be construed realistically and which merely instrumentally. The theory itself is mute – we must speak on its behalf.* » (Humphreys, 2004, p. 85)

Nous aurons l'occasion de revenir sur la question du réalisme sélectif (section 1.4.2). Tour-nous-nous plutôt, pour l'instant, vers une autre critique.

1.3.3 L'opacité épistémique

Cette dimension instrumentale, même partielle, de la conception de Humphreys prête le flanc à une autre critique, qui s'adresse surtout aux simulations informatiques, et concerne l'OPACITÉ ÉPISTÉMIQUE (*epistemic opacity*) :

« *In many computer simulations, the dynamic relationship between the initial and final states of the core simulation is epistemically opaque because most steps in the process are not open to direct inspection and verification. This opacity can result in a loss of understanding [...]* » (Humphreys, 2004, p. 148)

Humphreys distingue deux sources d'opacité épistémique. D'une part, les processus computationnels peuvent être tout simplement trop rapides pour qu'un humain en suive le flot. Ainsi, de tels processus sont transparents *en principe* (si un humain était éternel ou considérablement plus intelligent, il pourrait en comprendre le déroulement) mais opaques *en pratique*.

D'autre part, ils peuvent être ce que Wolfram (2002, p. 737-50) appelle des « *computationally irreducible processes* », des processus dont les interactions constituent la description la plus simple qu'on puisse en faire : donc, des processus irréductibles à une forme close⁹. Il s'agit en d'autres termes de « *irreducibly empirical states* » (Humphreys, 2006, p. 599), des états d'un système qui ne peuvent être découverts que par des voies empiriques. Outre les PROCESSUS COMPUTATIONNELLEMENT IRREDUCTIBLES, Humphreys (2006, p. 599) donne aussi l'exemple des phénomènes d'émergence faible : une propriété *P* d'un système *S* est faiblement émergente si elle n'est dérivable de la micro-structure de *S* que par simulation (Bedau, 2003, p. 12).

Cette opacité épistémique, d'une source ou d'une autre, constitue un défi auquel doit répondre la conception de Humphreys. Ce dernier voit dans le processus de construction – et la légitimation qu'il apporte au modèle – la solution :

« *The virtues of computational models [...] can be achieved by trial-and-error procedures treating the connections between the computational template and its solutions as a black box. There is nothing illegitimate about this as long as the motivation for the construction process is well founded.* » (Humphreys, 2004, p. 150)

En d'autres termes, la justification apportée par le processus de construction permettrait de mitiger les effets délétères de l'opacité épistémique. De même, le contrôle du modèle opaque (Lenhard, 2006a,b), p. ex. par l'ensemble de correction ou encore par la variation des paramètres de la simulation, permet un certain accès au phénomène, « *understanding in the pragmatic sense* » (Lenhard, 2006b, p. 609).

Lenhard (2006a) présente trois autres raisons qui permettent de croire que l'opacité épistémique ne constitue pas un problème insurmontable à la conception de Humphreys.

En premier lieu, il souligne que d'autres conceptions ou domaines ont réussi à surmonter un problème d'opacité :

« *A sibling of this question has been at the core of early cybernetics and information theory. [...] Quite significantly, scientists like C. Shannon, N. Wiener and A.N. Kolmogorov introduced a conception of information that was tailored for black boxes and for objects that can be measured only statistically.* » (Lenhard, 2006a, p. 2-3)

En second lieu, il faut compter avec le rôle rassurant de la représentation de sortie (la sixième et dernière composante de la définition formelle d'un modèle computationnel ; voir p. 15). En rendant visible – littéralement – les comportements du modèle, la représentation de sortie permet d'exercer un certain contrôle même si le fonctionnement demeure opaque. Ainsi,

⁹Cette formulation semble équivalente aux théories de type 2 de David Marr (Humphreys, 2006, p. 599). Une telle théorie prédit l'évolution d'un processus qui constitue lui-même sa propre description la plus simple.

« [a] fundamental ingredient of this methodology is a feedback loop between simulation experiments and visualizations of results that can render the model's dynamics tangible while it involves black-box elements. » (Lenhard, 2006a, p. 3)

En dernier lieu, Lenhard (2006a, p. 4) souligne le caractère anthropocentrique et subjectif de l'opacité épistémique. Humphreys propose à cet égard un exemple éclairant :

« The proofs that led to the classification of finite simple groups originally ran to a total of 15,000 pages, and part of the classification of the sporadic groups was completed with the aid of computers. If, contrary to fact, we were equipped with much faster brains and much longer life spans, this computational help would have been unnecessary. » (Humphreys, 2004, p. 46)

Or, l'empirisme traditionnel accepte déjà l'utilisation d'instruments dont le fonctionnement peut demeurer opaque à l'utilisateur :

« There exists a wonderful photograph of Ernest Rutherford's laboratory that nicely illustrates why theoretical knowledge can be unnecessary. Rutherford's voice was so loud that researchers were forced to hang a sign reading TALK SOFTLY PLEASE to prevent sound waves affecting their experiments. No fancy theory of sound was needed – the experimenters simply knew from experience that when Rutherford was around, their experiments went awry. » (Humphreys, 2004, p. 41)

Humphreys ne fait que réclamer la même courtoisie envers les instruments computationnels, proclamant du même coup que « the era of simple, anthropocentric mathematics is over » (Humphreys, 2004, p. 57).

1.3.4 Un pis-aller temporaire

Mais est-ce que cette « ère des mathématiques anthropocentriques » est bel et bien terminée, ou ne connaît-elle finalement qu'un déclin passager ?

Considérons un problème P solutionné présentement par un patron computationnel T . Or, nul part Humphreys (2004) ne défend l'idée que, parce que T est la seule solution actuellement disponible, jamais on ne pourra trouver de solution analytique (des forme close) à P . Cette idée est indéfendable pour les mêmes raisons que la conception de Humphreys est défendable : la même puissance computationnelle sans cesse croissante qui permet des modèles computationnels de plus en plus raffinés, permet également des moyens de plus en plus adéquats pour découvrir des solutions analytiques – donc, *non* computationnelles – à un problème.

En d'autres termes, on peut accepter la conception de Humphreys, tout en soulignant que les patrons et modèles computationnels sont utiles *temporairement*, jusqu'à ce que des solutions analytiques soient découvertes ou inventées, peut-être par des moyens computationnels. Une telle critique ne réfute en rien la conception de Humphreys, mais elle en limite quelque peu la portée. Gardons donc à l'esprit l'idée que la science computationnelle dont Humphreys (2004) défend l'existence et l'importance, n'est peut-être qu'une phase transitoire.

1.3.5 Hors-la-loi ?

Nous avons vu qu'Humphreys considère que la seconde « loi » du mouvement de Newton est en fait un patron théorique. Mais alors, qu'est-ce qui constitue une loi scientifique ? Humphreys semble disposé à éviter le débat :

« Often included as part of a theory, or considered separately, are scientific laws, generalizations that apply to and constrain the objects that form the subject matter of a field. Some examples of laws are [...] Newton's three laws [...]. It is notoriously difficult to provide criteria that distinguish scientific laws from other general statements which appear in theories. In what follows, we do not need to have a solution to that difficulty, for computational science can be highly successful without the need to refer to laws. » (Humphreys, 2004, p. 58)

Ainsi, selon Humphreys (2004) :

1. La seconde « loi » de Newton, malgré les guillemets, est bel et bien une loi, c'est-à-dire une généralisation qui s'applique à tous les objets du domaine – à tous les corps physique, dans le contexte de la physique newtonienne.
2. La différence entre une loi et une autre forme de généralisation est difficile à établir.
3. Dans le contexte de la science computationnelle, on peut se passer de faire référence à une loi. Nous avons vu (section 1.2.3) que les suppositions de construction peuvent se composer de principes, d'idéalisations, d'approximations et d'analogies. Le point important est qu'il n'est pas nécessaire qu'une des suppositions de construction soit une loi : l'exemple du patron du processus de Poisson (section 1.2.2) le montre bien.

Si Humphreys ne se rallie pas à la conception de Cartwright (1983), le résultat demeure le même : il ne sera pas – ou peu – question de loi scientifique en science computationnelle. Car s'il admet que les lois existent¹⁰, il considère qu'elles ne constituent pas l'unité d'analyse (l'*explanans* de la section 1.2, p. 12) adéquate à la science computationnelle : ce qui est important n'est pas le statut nomologique mais plutôt computationnel (tractable) d'une équation (Humphreys, 2004, p. 67). Ainsi, l'objectif premier de la science computationnelle n'est pas de découvrir des régularités nomologiques dans la nature, mais plutôt de construire des patrons (et modèles) computationnels tractables, en se fondant sur des principes qui *peuvent* (ou non) inclure des lois scientifiques.

1.4 Conséquences de l'empirisme computationnel

L'adoption de l'empirisme computationnel de Humphreys comporte un certain nombre de conséquences et pour la science, et pour la philosophie des sciences. Nous examinerons ce qui nous semble être les principales : l'accent mis sur ce qui est possible en pratique plutôt qu'en principe ; l'adoption d'un réalisme sélectif ; et enfin, la réorganisation possible des sciences.

¹⁰ « The point is not that there are no scientific laws – there are. [...] [T]here are laws but [they] need not be the starting point for applying science to the world. » (Humphreys, 2004, p. 89).

1.4.1 En pratique plutôt qu'en principe

La conception que propose Humphreys place la tractabilité au cœur des préoccupations de modélisation : un modèle, pour pouvoir être applicable et donc utile, doit être tractable, quitte à perdre en réalisme. Ainsi, de deux modèles non isomorphes représentant le même phénomène, on privilégiera celui qui est le plus tractable¹¹. Cela rejoint le premier constat (section 1.1.1, p. 1) : ce sont les mathématiques *tractables* qui contribuent au progrès de la science.

En particulier, l'empirisme computationnel de Humphreys évacue l'idée que les méthodes computationnelles n'ont rien de spécial ou de nouveau, puisqu'un humain sans aide extérieure pourrait, en principe, effectuer les mêmes calculs. Humphreys rapporte qu'un humain moyen calcule à une vitesse avoisinant 10^{-2} flops¹². Or, au moment d'écrire ces lignes, le superordinateur le plus performant sur Terre¹³ est capable de maintenir une vitesse de calcul de 280,6 téraflops (c'est-à-dire $280,6 \cdot 10^{12}$ flops). Ainsi, en 15,6 secondes, un seul superordinateur accomplira *grosso modo* le même nombre d'opérations qu'un humain moyen durant un laps de temps correspondant à l'âge de l'Univers (13,9 milliards d'années). Ce genre de comparaison fait dire à Humphreys :

« This extrapolation of our computational abilities take us to a region where the quantitatively different becomes the qualitatively different [...]. Speed matters. »
(Humphreys, 2004, p. 53)

Modulo l'emploi de certaines méthodes mathématiques nouvellement découvertes ou inventées, les opérations réalisées présentement avec des ordinateurs ont toujours été en principe à la portée de l'humain moyen doté d'une puissance de calcul de 10^{-2} flops. Or, il faudrait être de mauvaise foi – *« to put it mildly, disingenious »* (Humphreys, 2004, p. 52) – pour prétendre que l'avènement de l'ère informatique n'a pas entraîné de changement de fond en science. On doit donc cesser de ne s'intéresser qu'à ce qui est possible en principe, puisque ce sont les considérations pratiques qui, à toute fin utile, constituent les limites de la science. Ceci aura bien sûr des répercussions sur le type de réalisme scientifique que l'on peut adopter.

1.4.2 Un réalisme sélectif

L'idée que la conception de Humphreys n'est pas instrumentaliste a été explorée ci-dessus (section 1.3.2, p. 18). Humphreys lui-même prône une position non instrumentaliste, sans pour autant être un réaliste pur :

« To emphasize the importance of solvability within these computational models is not to adopt instrumentalism. Overall, such models are almost always false because

¹¹Cependant, Humphreys (2004) ne se prononce pas sur la question de choisir entre deux modèles non isomorphes de tractabilité égale.

¹²Un flop (*floating point operation per second*) est une unité de mesure correspondant à une opération en virgule flottante – addition ou multiplication de deux nombres comptant chacun treize chiffres – par seconde. Un humain moyen mettrait 100 secondes à effectuer une telle opération, d'où une vitesse de calcul de 10^{-2} flops (Humphreys, 2004, p. 52).

¹³Il s'agit du BlueGene/L situé au Lawrence Livermore National Laboratory, du ministère américain de l'Énergie (voir <http://www.top500.org/lists/2006/11> pour plus de détails).

of the presence of the correction set. But by asserting their falsity, we have already rejected an instrumentalist interpretation. Moreover, parts of the model often are true and some of the terms do refer. This is why one needs to be a selective realist. » (Humphreys, 2002, p. S7)

Résumons : la conception de Humphreys n'est ni instrumentaliste (puisqu'elle admet sans difficulté la fausseté des modèles), ni entièrement réaliste (puisqu'elle prône la primauté de la tractabilité sur la véridicité). Cette conception met donc en oeuvre une forme de réalisme partiel, le RÉALISME SÉLECTIF. En fait, selon Humphreys, le réalisme, en pratique, n'est que rarement pur : *« realists are rarely so unselective about, or so distanced from, the contents of their theories »* (Humphreys, 2004, p. 83). Le réalisme serait en effet une attitude qu'a un individu (ou groupe d'individus) envers des parties d'une théorie.

Une telle attitude de réalisme sélectif permet d'alléger la tension que nous avons souligné entre l'exactitude et la tractabilité, très proche de la tension entre la transparence épistémique et le réalisme. Le réalisme sélectif constitue donc le compromis qui s'impose :

« The obvious solution is to be explicit about which parts of the theory are to be construed realistically and which merely instrumentally. The theory itself is mute – we must speak on its behalf. » (Humphreys, 2004, p. 85)

1.4.3 La réorganisation des sciences

Une conséquence possible, mais certes pas nécessaire, de l'empirisme computationnel de Humphreys est la réorganisation des sciences, selon les patrons computationnels utilisés :

« Rather than science being organized in terms of various subject matters such as physics, biology, and sociology, it is now common for areas such as evolutionary biology and economics to share models because the modeling methods transcend the subject matter. » (Humphreys, 2007, p. 8)

Alors que le sujet concret constituait, pour l'expérimentaliste, le critère « naturel » pour diviser la science en disciplines distinctes, le patron computationnel, commun à plusieurs disciplines traditionnelles, peut remplir le même rôle dans cette ère de la science computationnelle. Et dans une certaine mesure, c'est déjà le cas : la science de la complexité, dont le Santa Fe Institute¹⁴ constitue certes l'un des phares, s'intéresse au patron commun à des domaines (physique théorique, biologie évolutionnaire, macroéconomie) pourtant très différents.

1.5 La nature des simulations informatiques

Un sous-domaine important de la science computationnelle est celui des simulations informatiques. Nous considérerons d'abord la place que leur réserve Humphreys, avant de s'attarder sur une critique de la position de celui-ci.

¹⁴Pour plus de détails, consulter <http://www.santafe.edu/>.

1.5.1 Définition et statut de la simulation informatique

Humphreys (2004) propose une définition de la simulation informatique qui fait intervenir deux notions importantes qu'il juge centrales. D'une part, la simulation produit un processus temporel dynamique. D'autre part, la simulation est un dispositif concret de calcul (*concrete computational device*).

Soit S un système. S est une SIMULATION BASIQUE (*core simulation*) d'un objet ou processus B si et seulement si (1) S est un dispositif concret de calcul qui (2) produit par un processus temporel dynamique (3) des solutions à un modèle computationnel M qui lui-même (4) représente, de manière statique ou dynamique, l'objet ou processus B . De plus, si le modèle computationnel M représente la structure du système réel R dans lequel s'inscrit B , alors S est une simulation basique de R par rapport à B (Humphreys, 2004, p. 110).

Enfin, si les résultats produits par S sont triés dans l'ordre qu'ils se manifestent dans le phénomène simulé, alors S est non seulement une simulation basique mais également une SIMULATION COMPLÈTE (*full simulation*) (Humphreys, 2004, p. 111).

Malgré cette définition précise, le statut exact des simulations dans la conception de Humphreys demeure nébuleux. Il est vrai que dans la définition présentée ci-dessus, Humphreys (2004) précise que les simulations sont des dispositifs concrets de calcul. Est-ce donc dire qu'il s'agirait d'instruments ? Rien n'est moins sûr, car Humphreys, dans la même oeuvre, fait plusieurs allusions à d'autres statuts possibles. Ainsi, une simulation informatique serait :

- Une expérience (numérique). « *[Simulations] are also used when [...] numerical experiments are more appropriate than empirical experiments.* » (Humphreys, 2004, p. 107). Aussi : « *Computer simulations, unlike empirical experiments, are precisely replicable [...] The experiment can be repeated exactly [...]* » (Humphreys, 2004, p. 115). Enfin, on note qu'Humphreys a déjà défendu l'idée que les simulations de Monte Carlo constituent des expériences (Humphreys, 1994).
- Un modèle. « *Simulations rely on an underlying computational model or are themselves models of a system* » (Humphreys, 2004, p. 107). Également : « *there is no difference between simulations and any other kind of theoretical model* » (Humphreys, 2004, p. 115).
- Un instrument. Outre la définition ci-dessus, qui présente la simulation comme étant un « dispositif concret », deux autres passages pointent dans cette direction : « *The computational process [of the core simulation] is thus an amplification instrument* » (Humphreys, 2004, p. 110). Et aussi : « *computational devices are the numerical analogues of empirical instruments* » (Humphreys, 2004, p. 116).
- Une autre sorte d'entité. « *[...] simulations can occupy the ground between theory and though experiments* » (Humphreys, 2004, p. 116).

Le moins qu'on puisse dire, c'est que la confusion règne ! Surtout que les autres oeuvres de Humphreys (par exemple (Humphreys, 1990)) ne sont pas d'une immense aide en la matière. Seule certitude : Humphreys considère que les simulations informatiques ne sont pas que de simples méthodes numériques. À cet égard, il fournit trois raisons de ne pas identifier les

premières aux mathématiques numériques (Humphreys, 2004, p. 112) :

- Au coeur d'une simulation informatique se trouve toujours un modèle computationnel, ce qui n'est pas nécessairement le cas d'une méthode numérique.
- Un élément computationnel concret – et donc un cortège de limites et contraintes concrètes – caractérise la simulation informatique mais pas les mathématiques numériques en général.
- Une simulation informatique doit fournir une représentation de sortie – graphique ou autre – afin de rendre les résultats intelligibles à l'humain, un problème qui ne touche pas toutes les méthodes numériques.

Cette caractérisation par la négative est certes utile mais ne permet pas d'alléger la confusion signalée ci-dessus, quant au statut des simulations informatiques. Néanmoins, nous croyons possible de cerner ce statut, en expliquant la confusion qui semble obscurcir la question, tout en respectant la définition donnée par Humphreys (2004). Notre approche repose en trois points :

- Considérons la simulation informatique comme étant un dispositif concret, ainsi que nous y invite la définition. Ce dispositif, tout comme les instruments empiriques (p. ex. le microscope), constitue un adjuvant épistémique concret à nos capacités naturelles : en bref, un *instrument*. Cette interprétation est d'ailleurs cautionnée à plusieurs reprises par Humphreys, par exemple lorsqu'il est question des « *instrumentally enhanced mathematics* » (Humphreys, 2004, p. 5-6).
- Cet instrument est fondé sur un *modèle* computationnel et peut même l'être lui-même, tout comme un astrolabe constitue un modèle de la voûte céleste, tout en étant également un instrument (et même, un instrument computationnel, quoique rudimentaire).
- Enfin, l'utilisation d'une simulation constitue une exploration « en pratique » – donc, entachée de contraintes concrètes telles que la puissance du processeur ou la taille de l'espace mémoire – du modèle sous-jacent, dont elle produit des solutions : en bref, une *expérience*. D'ailleurs, la simulation informatique, en exigeant l'utilisation de méthodes d'analyse de données (p. ex. forage de données, techniques statistiques, visualisation) ainsi que le souci constant de contrôler l'incertitude et l'erreur (on pense à l'ensemble de correction), porte l'empreinte méthodologique de l'expérimentation (Winsberg, 2003, p. 111).

En d'autres termes : la simulation informatique est un instrument servant à explorer expérimentalement les propriétés d'un modèle computationnel, une « *explorative cooperation between experimenting and modeling* » (Lenhard, 2007, p. 178). Ce qui se rapproche de la « *working definition* » que Humphreys (1990) proposait¹⁵ :

« *A computer simulation is any computer-implemented method for exploring the*

¹⁵Quatorze ans plus tard, Humphreys (2004, p. 107-108) juge que cette définition est à la fois trop large (car elle comprend des éléments de science assistée par ordinateur qui ne sont pas des simulations) et trop étroite (car il existe de véritables simulations qui portent pourtant sur des problèmes analytiquement solubles). Les deux problèmes, heureusement, n'ont pas d'impact sur notre discussion : toute critique à l'encontre de cette définition s'appliquera tout de même aux simulations informatiques de problèmes analytiquement insolubles, dont notre cas d'étude, la simulation d'Axelrod (1984, 1997).

properties of mathematical models where analytic methods are unavailable » (Humphreys, 1990, p. 501)

On peut se demander si ces méthodes analytiques ne sont pas disponibles (« unavailable ») en principe ou seulement en pratique. Humphreys ne répond pas directement à la question. Cependant, lorsqu'il compare les simulations informatiques aux méthodes numériques, il met au nombre des points communs ceci : « *[t]o provide solution methods for mathematical models where analytical methods are presently unavailable* » (Humphreys, 1990, p. 502; nous soulignons). Une disponibilité éventuelle d'une solution analytique n'est donc pas exclue, ce qui renforce la critique effectuée à la section 1.3.4 : la science computationnelle n'est peut-être que temporaire.

Ayant établi de manière plus précise le statut des simulations informatiques dans la conception de Humphreys, nous pouvons maintenant confronter à la critique ce concept de simulation *qua* instrument expérimental, en nous appuyant sur la position de Hacking (1981, 1989), qui s'est intéressé longuement au rôle des instruments en science.

1.5.2 La simulation *qua* instrument expérimental

Notre critique de la notion de simulation en tant qu'instrument – et de son utilisation en tant qu'expérience – comportera deux moments. En premier lieu, nous nous attarderons sur l'IMPRÉGNATION THÉORIQUE (*theory-ladenness*) de la simulation informatique. En second lieu, nous nous intéresserons à la justification de cette dernière.

Rappelons que la simulation, par définition (présentée ci-dessus), est toujours fondée sur un modèle computationnel¹⁶. Or ce modèle est très souvent – et peut-être toujours – fondé sur des principes théoriques. Ainsi, par construction, la simulation informatique comprend – fréquemment – une certaine imprégnation théorique. Évidemment, il y a des exceptions : le modèle de Poisson (section 1.2.2) est un tel exemple de modèle computationnel ne reposant sur aucune loi ou principe théorique.

Peu importe, car notre critique ne porte pas sur cette imprégnation théorique par construction, mais plutôt sur l'imprégnation théorique liée à l'utilisation de la simulation en tant qu'instrument. En effet, selon Humphreys, pour utiliser la simulation en tant qu'instrument, il faut souvent – mais pas toujours – avoir recours à une connaissance du fonctionnement de la simulation, connaissance qui *peut* être théorique :

« You do, as we have seen, often need to know how the instrument works in order to use it effectively, but this knowledge is not always based on theory or, even when it is, upon correct theory. » (Humphreys, 2004, p. 40)

Or, *a priori*, cette affirmation est loin d'aller de soi. En effet, Hacking (1981) s'oppose à l'idée que la connaissance du principe de fonctionnement d'un instrument – dont, hypothétiquement, la simulation informatique – soit nécessaire à son utilisation :

¹⁶En fait, Lenhard (2007) signale à juste titre qu'une simulation informatique d'un phénomène naturel est basée sur la modélisation discrète d'un modèle mathématique de ce phénomène, puisque l'ordinateur ne peut manipuler qu'un modèle discret plutôt que continu. Ainsi, toute simulation requiert deux efforts successifs de modélisation.

« *One needs theory to make a microscope. You do not need theory to use one. Theory may help to understand why objects perceived with an interference-contrast microscope have asymmetric fringes around them, but you can learn to disregard that effect quite empirically.* » (Hacking, 1981, p. 309)

S'il est question ici de microscopie et non de simulation informatique, la critique reste valide : Humphreys, après tout, insiste à plus d'une reprise sur l'analogie entre les méthodes empiriques et computationnelles, allant jusqu'à utiliser la métaphore du microscope pour expliquer la simulation informatique¹⁷.

Hacking (1981) rejette la nécessité de connaître le fonctionnement d'un instrument pour pouvoir l'utiliser, par trois arguments :

- L'observation à l'aide d'un instrument n'est pas simplement passive, mais plutôt une interaction : « *Don't just peer : interfere* » (Hacking, 1981, p. 308). L'apprentissage de l'utilisation d'un instrument ne passerait donc pas par la connaissance théorique de son fonctionnement, mais plutôt par la pratique. Hacking, évidemment, concède que la connaissance théorique peut quand même être utile (Hacking, 1981, p. 317).
- La théorie peut aider à comprendre la présence d'artefacts de l'instrument, mais elle n'est pas nécessaire : cela pourrait se faire empiriquement.
- L'ARGUMENT DE LA COÏNCIDENCE (plusieurs instruments différents révélant la même structure, celle-ci doit être réelle ; sinon, la coïncidence serait extraordinaire) ne nécessite pas de connaissance théorique pour s'appliquer : sans connaître la nature de l'entité *E*, il est possible de conclure qu'elle est réelle simplement parce qu'elle est révélée par plusieurs instruments fonctionnant selon des principes différents¹⁸ (Hacking, 1981, p. 315), c'est-à-dire sans avoir à se prononcer sur ce qu'est *E*.

Ces trois arguments se conjuguent pour remettre en doute l'idée que la connaissance du fonctionnement d'un instrument est nécessaire à son utilisation.

Humphreys (2004) répond à cette critique en présentant plusieurs exemples (p. ex. imagerie par résonance magnétique) où la connaissance du fonctionnement est nécessaire pour comprendre l'origine d'observations aberrantes. En définitive,

« *the theory of how the instrument operates is routinely needed to protect the user against skeptical objections resulting from ways in which the instrument can produce misleading outputs.* » (Humphreys, 2004, p. 38)

Nous avons jusqu'à maintenant séparé l'imprégnation théorique provenant de la construction de celle qui semble nécessaire à l'utilisation de la simulation informatique. Mais peut-être n'y a-t-il pas lieu de maintenir cette étanchéité. En effet, le monde simulé est entièrement déterminé par un formalisme mathématique ou computationnel (le modèle computationnel sous-jacent), qui agit comme théorie régissant le fonctionnement de ce monde. Ainsi, toute simulation, en tant qu'instrument, est donc imprégnée de ces principes de fonctionnement – théoriques

¹⁷Voir par exemple (Humphreys, 1995a, p. 126), (Humphreys, 2000) et (Humphreys, 2004, p. 105).

¹⁸Mais on pourrait avancer l'idée que la connaissance du fonctionnement de ces instruments *est* nécessaire pour qu'on puisse établir qu'ils sont fondés sur des principes différents...

– du monde, puisque c'est ce dernier dont elle doit rendre compte. Donc, les mêmes principes théoriques au fondement du monde (simulé) sont aussi les principes que l'on doit connaître pour utiliser adéquatement la simulation *qua* instrument.

Résumons : *contra* Hacking, il semble qu'une imprégnation théorique soit souvent présente lors de l'utilisation d'une simulation informatique, d'une part pour repérer et expliquer la présence d'artefacts et d'autre part, parce que cette utilisation ne peut se faire que dans un monde entièrement déterminé par un modèle computationnel et donc, en dernière analyse, un patron *théorique*.

Ceci nous entraîne naturellement vers notre deuxième point de critique : la question de la justification de la simulation informatique. Qu'est-ce qui justifie les résultats obtenus par la simulation informatique ?

Le problème est de taille. La simulation informatique, fondée sur un modèle computationnel, comporte par définition une représentation de sortie, souvent graphique : un diagramme, une image, voire une vidéo. Le danger de ce réalisme au sens commun du terme, est qu'il peut provoquer « *the illusion of veridicality [made] even more compelling by the presentation of these three-dimensional images in time.* » (Keller, 2002, p. 224). La question de la justification d'une simulation est donc importante.

Dans la conception de Humphreys (2004), la justification d'une simulation émane du modèle computationnel sous-jacent. En effet, nous avons vu qu'à tout modèle – et même, tout patron – computationnel correspond une justification initiale provenant du processus de construction (voir la section 1.2.3, p. 13). Les résultats de la simulation sont justifiés par le fait même qu'ils constituent des solutions à un modèle computationnel lui-même justifié par son propre processus de construction.

Mais si nous considérons la simulation informatique en tant qu'instrument, une autre forme de justification apparaît. Pour Hacking (1981), l'utilisation répétée et continue de l'instrument constitue une autre source de justification :

« *Practice – and I mean in general going [sic], not looking – creates the ability to distinguish between visible artefacts of the preparation or the instrument, and the real structure that is seen with the microscope. This practical ability breeds conviction.* » (Hacking, 1981, p. 309)

En d'autres termes, la justification émergerait non seulement de la construction (du modèle computationnel sous-jacent) mais aussi de l'utilisation (de la simulation *qua* instrument).

Une autre source de justification de la simulation informatique est « l'histoire de vie » des expériences :

« *Over a decade ago I wrote that experiments have a life of their own. I intended partly to convey the fact that experiments are organic, develop, change, and yet retain a certain long-term development which makes us talk about repeating and replicating experiments [...]. I think of experiments as having a life : maturing, evolving, adapting, being not only recycled but also, quite literally, being retooled.* » (Hacking, 1992, p. 307)

La simulation informatique constitue un exemple criant de ce dynamisme : la facilité avec laquelle elle peut, à titre de programme informatique, être remaniée, améliorée, adaptée, etc., constitue même l'un de ses avantages (Humphreys, 2004, p. 114-6).

Ainsi, une simulation informatique donnée (p. ex. une simulation d'automates cellulaires, comme le « Game of Life » de John Conway) a un cycle de vie avec, à la clé, la maturité et l'autorité d'une expérience – ou d'un instrument – bien établie. En d'autres termes,

« The history of a simulation technique is very much like the history of a scientific instrument. It begins with a relatively crude and simple technique for attacking a relatively small set of problems. Over time, the instrument or technique is called upon to attack a larger set of problems or to achieve a higher degree of accuracy. »
(Winsberg, 2003, p. 123-4)

Fait à noter : l'amélioration d'une simulation informatique peut provenir de l'ensemble de correction du modèle computationnel sous-jacent. La construction de ce dernier contribue donc à l'enrichissement éventuel de la simulation et donc de sa justification.

Pour résumer, la justification d'une simulation provient, non seulement de sa construction (Humphreys), mais également de son utilisation (Hacking) et de sa réputation¹⁹ (Hacking, Winsberg).

La confrontation des conceptions respectives de Humphreys (2004) et de Hacking (1981) nous a donc permis de confirmer la position du premier sur la question de l'imprégnation théorique : la théorie est souvent nécessaire pour utiliser la simulation informatique *qua* instrument. Par contre, nous avons pu mettre à contribution la position de Hacking sur la question de la justification de la simulation informatique : la justification provient, non seulement de la construction, mais également de l'utilisation et de la réputation de l'instrument. Nous sommes maintenant en mesure de confronter la théorie à la pratique, en tentant d'appliquer la conception de Humphreys à une simulation informatique particulière : l'expérience d'Axelrod (1984, 1997).

¹⁹Le traitement superficiel qui a été fait de ce dernier point ne rend sûrement pas justice à la complexité du concept de réputation en science. Mais une telle question est d'une part moins centrale à notre propos et, d'autre part, relève plus de la sociologie que de la philosophie des sciences. Le lecteur intéressé par la question est prié de consulter un bon ouvrage sur la question, p. ex. (Merton, 1973).

Chapitre 2

La simulation d'Axelrod

Dans un premier temps, nous présenterons brièvement l'expérience effectuée par Axelrod (1984). Puis, nous nous efforcerons de dégager de cette simulation la structure de modèle computationnel de Humphreys (2004). Enfin, nous mettrons en évidence certains points de convergence et de divergence entre la conception de la simulation informatique de Humphreys (2004) et la structure effectivement révélée par notre cas d'étude. Le tout nous permettra d'améliorer notre compréhension – mais aussi, notre critique – de la conception de Humphreys (2004), ainsi que d'éclairer le rôle épistémique que peuvent jouer des simulations informatiques comme celles utilisées par Axelrod, dans le domaine de la biologie évolutionnaire.

2.1 Présentation de l'expérience

Imaginons deux personnes arrêtées par les autorités et emprisonnées séparément. À chacun des prisonnier, on propose le marché suivant :

- si un – et un seul – des deux prisonniers accepte de témoigner contre son complice, alors le juge se montrera clément envers le délateur mais sévère envers le prisonnier plus loyal ;
- si les deux prisonniers témoignent l'un contre l'autre, alors ils recevront *tous les deux* une peine exemplaire, leur culpabilité ne faisant aucun doute ;
- cependant, si aucun des prisonniers ne témoigne contre son complice, le juge, ne disposant que de preuves circonstancielle, ne pourra les condamner qu'à une peine légère pour des chefs d'accusation accessoires.

Tel est le dilemme du prisonnier : bien qu'il soit toujours plus avantageux – et donc, rationnel – pour un prisonnier de témoigner contre son complice, il reste que la délation mutuelle (le second cas) est moins avantageuse que la coopération (le troisième cas).

2.1.1 Le dilemme du prisonnier

Formalisons la situation présentée. Le DILEMME DU PRISONNIER est un jeu à deux participants, servant couramment à modéliser les interactions entre deux agents, et ce dans plusieurs

	Il coopère	Il fait défection
Je coopère	Nous recevons chacun $R = 3$ (la récompense de coopération mutuelle).	Je reçois $S = 0$ (le « sucker's payoff »), il reçoit $T = 5$ (la tentation de défection).
Je fais défection	Je reçois $T = 5$ (la tentation de défection), il reçoit $S = 0$ (le « sucker's payoff »).	Nous recevons chacun $P = 1$ (la punition de défection mutuelle).

TAB. 2.1 – Le dilemme du prisonnier

disciplines¹. Le déroulement de ce jeu peut être présenté comme suit². Deux joueurs doivent, chacun à l'insu de l'autre, choisir de coopérer avec l'adversaire ou encore de faire défection. Après que les deux joueurs aient effectué leur choix, ils reçoivent un gain possiblement nul selon la matrice présentée au tableau 2.1.

Ces gains doivent nécessairement respecter deux inégalités. D'une part, $T > R > P > S$, c'est-à-dire qu'il est toujours plus avantageux de faire défection que de collaborer, peu importe le choix de l'adversaire. D'autre part, $R > \frac{T+S}{2}$, c'est-à-dire que les deux joueurs ne peuvent échapper au dilemme en s'exploitant mutuellement, à tour de rôle, dans une série d'interactions. Le dilemme tient au fait que, s'il est toujours plus avantageux pour un joueur de faire défection que de collaborer, la défection mutuelle est cependant moins avantageuse que la coopération – en d'autres termes, que $R > P$.

La situation se complexifie lorsque les deux joueurs sont mis en présence l'un de l'autre de manière répétée. Cette variante, appelée DILEMME ITÉRÉ DU PRISONNIER, permet alors à un joueur de baser son choix, non seulement sur la matrice des gains, mais également sur l'historique des interactions avec son adversaire. Il devient possible, par exemple, de « punir » l'adversaire (en faisant défection) pour une défection passée, ou encore de lui « pardonner » (en coopérant malgré la défection passée de l'adversaire).

Un exemple de partie du dilemme itéré du prisonnier est donné au tableau 2.2. Alors que le joueur A utilise la stratégie TIT FOR TAT (sur laquelle nous reviendrons à la fin de la section 2.1.2), le joueur B utilise une « stratégie » aléatoire : ses choix sont le résultat du lancer d'une pièce de monnaie. La partie, après 10 itérations, se solde par un score égal pour les deux joueurs.

2.1.2 La simulation informatique

C'est sur ce dilemme itéré du prisonnier que porte la simulation d'Axelrod (1984). L'usage du singulier pouvant provoquer la confusion, nous nous efforçons de préciser qu'Axelrod

¹ « *The iterated Prisoner's Dilemma has become the E. coli of social psychology.* » (Axelrod, 1984, p. 28) Lorsqu'on connaît la place importante de cet organisme exemplaire en microbiologie, l'affirmation d'Axelrod prend tout son sens. Treize ans plus tard, Axelrod renchérit : « *[i]t has even become a standard paradigm for studying issues in fields as diverse as evolutionary biology and networked computer systems.* » (Axelrod, 1997, p. xi)

² Plusieurs auteurs présentent le jeu du dilemme du prisonnier. Nous reprenons ici la présentation qu'en fait Axelrod (1984, p. 7-9).

Itération	1	2	3	4	5	6	7	8	9	10	Total
Choix (joueur A)	C	C	C	C	D	D	C	C	C	D	
Score cumulatif (A)	3	6	9	9	10	15	18	21	21	26	26
Choix (joueur B)	C	C	C	D	D	C	C	C	D	C	
Score cumulatif (B)	3	6	9	14	15	15	18	21	26	26	26

TAB. 2.2 – Exemple du dilemme itéré du prisonnier (C : coopération ; D : défection)

a effectué un certain nombre de simulations (présentées dans (Axelrod, 1984, 1997)) portant toutes sur le dilemme itéré du prisonnier, mais chacune comptant des raffinements sur les précédentes. Puisque l'analyse de ces raffinements fait partie de notre application de la conception de Humphreys (2004), nous les passerons pour l'instant sous silence. La présentation qui suit se limite donc à la toute première expérience d'Axelrod (1980a).

Un tournoi met en présence quinze joueurs, chacun étant en fait une stratégie programmée dans un langage informatique. Quatorze de ces stratégies ont été fournies par des scientifiques de différentes disciplines : psychologie, sciences économiques, mathématiques, sciences politiques, etc. Il y a donc en principe une grande diversité dans les stratégies utilisées, qui vise à estomper les biais inhérents à chaque discipline. Le quinzième et dernier joueur, appelé « RANDOM », se contente de prendre ses décisions au hasard : à chaque interaction, RANDOM choisit aléatoirement son coup – coopérer ou faire défection. Il agit informellement comme joueur « témoin », voire comme étalon : bien mauvaise sera la stratégie qui aura des performances inférieures à celles de RANDOM !

Durant le tournoi, chaque joueur joue avec chacun des autres joueurs – ainsi qu'avec un « clone » de lui-même, utilisant la même stratégie – une partie comptant 200 interactions ³. Contrairement aux tournois conventionnels, le gagnant du tournoi n'est pas le joueur ayant battu chacun des autres joueurs, mais plutôt celui ayant obtenu, lors des différentes parties, le meilleur gain moyen par partie.

Lors de la première expérience d'Axelrod (1980a), le gagnant (avec 504,5 points) s'est avéré être le joueur TIT FOR TAT, dont la stratégie est :

- Au premier tour, collaborer.
- À chaque tour subséquent, effectuer le même choix que ce que l'adversaire a fait au coup précédent. En d'autres termes, si l'adversaire, à l'interaction k , a collaboré (respectivement fait défection), alors TIT FOR TAT, à l'interaction $k + 1$, collaborera (resp. fera défection).

Au terme de cette brève présentation des expériences premières d'Axelrod (1980a), nous sommes maintenant en mesure d'appliquer à cette simulation la conception des modèles computationnels de Humphreys (2004).

³De plus, le tournoi, afin d'assurer la stabilité des résultats, a été répété cinq fois. C'est donc $15 \cdot 15 \cdot 200 \cdot 5 = 225000$ interactions qui auraient été réalisées. L'origine du nombre de choix individuels rapporté à plusieurs reprises (Axelrod 1980a, p. 8 ; 1984, p. 31), à savoir 240000 (plutôt que $2 \cdot 225000 = 450000$), demeure un mystère.

2.2 Dégagement de la structure de modèle computationnel

La pièce maîtresse de cette application de la conception de Humphreys (2004) à la simulation d'Axelrod (1984) consiste à dégager de cette dernière la structure de modèle computationnel. Nous nous proposons d'atteindre cet objectif en trois moments : d'abord, identification du patron théorique et du patron computationnel ; puis, explicitation des suppositions de construction ; enfin, délimitation de l'ensemble de correction, avec illustration de la mise en oeuvre de cet ensemble dans le raffinement successif des expériences d'Axelrod. Nous serons alors en mesure de porter un regard critique sur la conception de Humphreys (2004).

2.2.1 Patron computationnel

La première étape de notre parcours doit être le développement du patron sous-jacent à la simulation d'Axelrod (1984). Il a été précédemment question (section 1.2.1) de la différence entre patron théorique, qui ne fait que préciser une relation générale entre entités, et patron computationnel, suffisamment spécifique pour être tractable. Dans cette section, nous allons donc développer le patron théorique du dilemme du prisonnier, avant de passer à un patron computationnel correspondant.

Notons d'abord que le développement de ce patron sera effectué selon une approche mathématique. Bien qu'Humphreys admette la possibilité qu'un patron puisse être défini « *in some specific computational language* » (Humphreys, 2004, p. 152), il ne présente, tout au long de son oeuvre, que des représentations *mathématiques* des patrons. Notons également qu'Axelrod (1984, 1997) lui-même ne propose pas de mathématisation appropriée de ses expériences ; ce qui suit ne doit donc pas être considéré comme étant une formulation canonique, mais plutôt un exemple de formalisation.

Remarquons enfin que le développement mathématique du patron théorique pourra sembler inutilement complexe. Il est possible, voire probable, que des approches mathématiques différentes et meilleures puissent être prises pour rendre compte du patron théorique du dilemme du prisonnier. Mais le but de la discussion n'est pas de montrer l'unicité ou l'optimalité d'une formulation mathématique particulière, mais plutôt de prouver, par l'exemple, qu'une formulation mathématique est *possible*, et donc qu'il existe bel et bien un patron théorique du dilemme du prisonnier. Le lecteur convaincu de cette possibilité d'existence est invité à passer à la section suivante (2.2.2, p. 35).

Passons maintenant au développement proprement dit. Le dilemme du prisonnier est un jeu entre deux joueurs A et B , comptant $n \geq 1$ interactions. La stratégie de chaque joueur – son « programme » – est représentée par une fonction prenant en compte deux paramètres : l'historique des choix du joueur lui-même, ainsi que l'historique des choix de l'adversaire (l'historique de chacun étant représentée par une chaîne de bits). Par exemple, la stratégie du joueur A à la k -ème interaction est représentée par la fonction $f_A(h_{A,k}, h_{B,k})$. Cette fonction prend la valeur 0 pour signifier une défection et la valeur 1 pour signifier une coopération. Quant au gain

effectué dans les mêmes circonstances, il est déterminé par la fonction $g(c_{A,k}, c_{B,k})$, où $c_{A,k}$ (respectivement $c_{B,k}$) représente le choix (donc, 0 ou 1), déterminé par f_A (resp. f_B), effectué par le joueur A (resp. B) à la k -ème interaction. Ce gain prend une valeur positive ou nulle. Par exemple, en prenant les valeurs présentées au tableau 2.1 (p. 31), nous obtenons la fonction

$$g(0, 0) = 1; g(0, 1) = 5; g(1, 0) = 0; g(1, 1) = 3$$

Ainsi, le score s_A obtenu par le joueur A lors d'une partie comprenant n interactions avec le joueur B est donné par la formule :

$$s_A(n) = \sum_{k=1}^n g[f_A(h_{A,k}, h_{B,k}), f_B(h_{B,k}, h_{A,k})]$$

et de manière similaire pour le joueur B . En d'autres termes, le score obtenu est la somme du gain obtenu à chaque interaction, gain lui-même déterminé d'une part par le choix effectué (défection ou coopération) par le joueur, d'autre part par le choix effectué par l'adversaire ; chaque choix étant fonction de l'historique des choix passés du joueur et de son adversaire.

Jusqu'à maintenant, le patron théorique présenté n'a rien de spécifique au cas d'étude et pourrait convenir à plusieurs jeux formels entre deux joueurs. Afin d'en faire véritablement le patron théorique du dilemme du prisonnier, nous devons donc spécifier les contraintes sur les gains. Ainsi,

$$g(0, 1) > g(1, 1) > g(0, 0) > g(1, 0)$$

c'est-à-dire que la défection est toujours plus avantageuse que la coopération, peu importe le choix effectué par l'adversaire. De même,

$$g(1, 1) > \frac{g(0, 1) + g(1, 0)}{2}$$

c'est-à-dire que les deux joueurs ne peuvent échapper au dilemme en s'exploitant mutuellement, à tour de rôle.

Quelques remarques supplémentaires sur ce patron théorique. D'une part, il ne faut pas que la notation masque le fait que la fonction représentant la stratégie est, ultimement, définie par récurrence. Ainsi, sa valeur à une itération dépend de la valeur qu'elle a prise à l'itération précédente ainsi que de la valeur prise par la fonction représentant la stratégie de l'adversaire à cette même itération précédente. Loin d'être une fonction « prédéfinie », elle a donc un certain développement dans le temps, un dynamisme, qui seront exprimés dans la simulation informatique (nous reviendrons sur cette caractéristique importante à la section 2.3).

D'autre part, l'objectif visé par les expériences d'Axelrod devrait maintenant être clair : trouver des solutions au patron, c'est-à-dire identifier des fonctions (stratégies) maximisant la moyenne du score s_A obtenu lors de différentes parties. Il s'agit donc, en quelque sorte, d'un problème d'optimisation.

Le passage du patron théorique à un patron computationnel peut se faire de manière aisée : en fixant le nombre d'interactions dans une partie (p. ex. $n = 200$, comme il a été fait lors de la première expérience (Axelrod, 1980a)) et en définissant les gains pour chaque couple de choix. Ainsi :

$$g(0, 0) = 1; g(0, 1) = 5; g(1, 0) = 0; g(1, 1) = 3$$

selon les mêmes gains que ceux présentés dans le tableau 2.1 (p. 31).

Il est clair que ce patron computationnel est tractable, puisqu'il a été utilisé tel quel et avec succès dans la première expérience d'Axelrod (1980a) : si le patron n'était pas tractable, l'expérience aurait tout simplement échoué. Le coeur du modèle computationnel étant en place, nous pouvons maintenant nous intéresser aux suppositions de construction.

2.2.2 Suppositions de construction

Nous avons présenté à la section 1.2.3 les suppositions de construction, une composante de tout modèle computationnel. Dans la présente section, nous dégagerons les différentes suppositions de construction du dilemme itéré du prisonnier, suivant Humphreys (2004) et son quintuple (ontologie, idéalisation, abstractions, contraintes, approximations) présenté à la section 1.2.3.

L'ontologie du dilemme itéré du prisonnier ne compte *a priori* que deux sortes d'entités : d'une part, des agents et d'autre part, des interactions entre agents. Les agents comportent quelques caractéristiques importantes : ils ont leur propre intérêt à coeur ; ils peuvent se reconnaître entre eux ou du moins limiter leurs interactions à un autre agent particulier⁴ ; enfin, ils peuvent, d'une façon ou d'une autre, tenir compte dans leurs « délibérations » de l'historique de leurs interactions avec l'adversaire (Axelrod, 1984, p. 11).

Fait notable : l'ontologie ne compte ni de loi, ni de principe de base. En effet, elle ne comporte que des agents et des interactions entre agents, c'est-à-dire des entités *définies* (plutôt que modélisées à partir de cas réels). Ainsi, ce patron du dilemme itéré du prisonnier est, tout comme le patron du processus de Poisson (1.2.2), un patron sans dimension nomologique, confirmant ainsi que « *[t]heories need play no role in constructing templates.* » (Humphreys, 2004, p. 89).

Avant de poursuivre, précisons que la différence entre idéalisation, abstraction, contrainte et approximation n'est pas toujours claire, et qu'un élément placé dans une rubrique pourrait bien appartenir à une autre. Pire, sauf dans le cas de l'abstraction, Humphreys (2004) ne fournit pas de définition ou explication claire sur ce qu'il entend par ces termes.

Heureusement, ce flou ne nuit pas à notre objectif. Puisque nous ne voulons qu'établir que l'expérience d'Axelrod est un modèle computationnel, il suffit de montrer que toutes les caractéristiques de l'expérience trouvent leur place dans le cadre conceptuel du modèle computationnel de Humphreys (2004) ; la place précise de chacune importe alors peu pour notre propos.

⁴Dans le cadre des diverses expériences d'Axelrod, cette supposition se manifeste dans le fait que, dans une partie donnée, un joueur *A* peut compter sur le fait que son adversaire demeure le même joueur *B* ; il n'y a pas de changement abrupt d'adversaire pendant une partie donnée.

Nous devons d'abord considérer les idéalizations sur lesquelles se fonde le patron du dilemme itéré du prisonnier. Une idéalisation évidente est le fait que les interactions, dans le patron, ne s'effectuent qu'entre deux joueurs.

Différentes approches ont été proposées pour modéliser le dilemme du prisonnier à n joueurs⁵ ; mais, sans doute pour des raisons de simplicité computationnelle, Axelrod (1984) se limite aux interactions entre deux joueurs : une idéalisation nécessaire, en effet, car « *[w]hen originally run in 1978, the round-robin tournament pushed the capabilities of a large mainframe computer* » (Axelrod, 1997, p. 32).

Une autre idéalisation, plus subtile, est l'absence de bruit (Wu et Axelrod, 1995). En effet, dans l'expérience originale (Axelrod, 1980a), le choix effectué par un joueur est toujours réalisé, sans erreur. Si un joueur choisit de coopérer (respectivement faire défection), alors avec une probabilité de 1 (100%) il coopérera (resp. fera défection). Contrastons avec une situation où il y aurait 1% de bruit. Dans ces conditions, un joueur choisissant de coopérer (resp. faire défection) pourrait tout de même, avec une probabilité de 0,01, faire défection (resp. coopérer). Le bruit est donc une entrave à la réalisation d'une décision, dont l'expérience originale d'Axelrod (1980a) fixe cependant la valeur à 0. Notons que l'absence de bruit est un exemple de caractéristique du modèle qui est difficile à catégoriser. S'agit-il, tel que nous le présentons ici, d'une idéalisation – c'est-à-dire qu'on fixe le bruit à un niveau nul (zéro) – ou d'une abstraction ? Peu importe, puisque dans les deux cas, cette caractéristique prend sa place dans le cadre des suppositions de construction proposé par Humphreys (2004).

Enfin, il y a une certaine idéalisation quant à la représentativité et la possibilité d'émergence des stratégies fournies par les participants : la première expérience d'Axelrod (1984) repose sur le fait que les quinze stratégies fournies par des participants (humains) sont représentatives de toutes les stratégies possibles (en d'autres termes, qu'il y a absence de biais). Dans les faits, il se pourrait que le choix des stratégies soit biaisé ou encore qu'une de ces stratégies reflète les particularités de la rationalité humaine de son concepteur.

En troisième lieu, deux abstractions⁶ sont mises en oeuvre. D'une part, la menace n'est pas possible : un joueur ne peut, par la menace, contraindre son adversaire à coopérer (ou à faire défection). D'autre part, un joueur n'a pas accès à l'historique des interactions de son adversaire *avec d'autres joueurs*. Le patron du dilemme itéré du prisonnier fait donc abstraction de tout concept de réputation du joueur.

En quatrième lieu, nous devons identifier les contraintes de construction. Une contrainte évidente est le fait que le nombre d'interactions par partie est fixé d'avance, p. ex. à 200 dans le cas de la première expérience (Axelrod, 1980a). Notons immédiatement une difficulté que soulève cette contrainte : un joueur qui sait que l'interaction courante est la dernière avec cet adversaire particulier n'a plus à craindre de représailles. Logiquement, tous les joueurs devraient

⁵Voir par exemple (Olson, 1965).

⁶Tel que vu à la note 7 (p. 14), nous utilisons la définition d'abstraction de Humphreys (2004, p. 143) : « *the process of omitting properties or their related predicates from a system or a model, the omission process being either cognitive, representational, or physical, leaving the remaining properties or predicates unchanged.* ».

faire défection à la dernière interaction ; par le même raisonnement, ils devraient aussi faire défection à l'avant-dernière, etc. ; par induction (mathématique), les joueurs devraient donc toujours faire défection ! Il y a ici un risque réel de ce qu'Axelrod appelle des « *end-game effects* » (Axelrod, 1984, p. 42-3).

En cinquième – et dernier – lieu, notons qu'il n'y a aucune approximation dans le patron du dilemme itéré du prisonnier. En effet, les seules valeurs numériques utilisées – les gains potentiels $g(c_{A,k}, c_{B,k})$ (p. 34) – sont fixées par décret : elles ne visent pas à représenter des valeurs existant dans la nature. Donc, à ce titre, elles ne peuvent être des approximations empiriques et doivent plutôt être considérées comme absolument précises, puisque fixées *par définition*.

Les cinq éléments des suppositions de construction ayant été identifiés, examinons maintenant comment elles trouvent écho dans l'ensemble de correction.

2.2.3 Ensemble de correction

Rappelons que pour Humphreys (2004), l'ensemble de correction (abordé à la section 1.2.3) renferme toutes les modifications aux suppositions de construction qui pourront être apportées « *when (not if)* » (Humphreys, 2004, p. 78) le patron computationnel déviara des résultats empiriques.

Or, dans le cas à l'étude, l'ensemble de correction n'est pas utilisé pour combler l'écart entre des données empiriques et le patron computationnel. En effet, ce dernier, à notre connaissance, n'est jamais confronté à l'aune de données réelles. Par contre, Axelrod (1984, 1997) utilise plusieurs éléments de l'ensemble de correction afin de raffiner le patron computationnel du dilemme itéré du prisonnier. Nous considérerons dans cette section trois exemples de raffinements issus de l'ensemble de correction ; puis nous reviendrons sur le statut de ces raffinements par rapport à l'empirie.

La seconde expérience d'Axelrod (1980b) introduit une amélioration par rapport à la première (Axelrod, 1980a). Outre un plus grand nombre (63 plutôt que 15) de joueurs, cette seconde expérience comporte une caractéristique importante : le nombre d'interactions par partie n'est plus fixé d'avance. En effet,

« [a]s announced in the rules the length of the games was determined probabilistically with a .00346 chance of ending with each given move. [...] Thus the average length of a game turned out to be somewhat shorter than expected at 151 moves. Since no one knew exactly when the last move would come, end-game effects were successfully avoided in the second round. » (Axelrod, 1980b, p. 383)

Ce raffinement probabiliste constitue selon nous un exemple de relaxation de contrainte (Humphreys, 2004, p. 79), un élément de l'ensemble de correction : en effet, la contrainte des 200 interactions, identifiée à la section 2.2.1, est maintenant levée.

Un deuxième exemple d'utilisation de l'ensemble de correction nous est donné par une expérience ultérieure d'Axelrod (1987) et concerne les stratégies utilisées par les joueurs. Les expériences initiales d'Axelrod (1980a,b) utilisent des stratégies fournies par les participants

(humains). De plus, ces stratégies demeurent inchangées (c.-à-d. qu'elles ne « s'adaptent » pas) tout au long de l'expérience. Or, on peut se demander d'une part si cela n'introduit pas un biais dans les premières expériences ; et d'autre part, si ces stratégies ne peuvent être conçues que par des agents entachés des particularités de la rationalité humaine. Le recours à l'algorithmique génétique est une amélioration visant à contrer ces deux objections, tout en introduisant une certaine capacité d'adaptation.

Brièvement⁷, un algorithme génétique est un algorithme qui obtient itérativement des solutions de plus en plus adéquates à un problème en combinant « génétiquement » les solutions les plus performantes de l'itération précédente. Étant donné un problème P , un algorithme génétique visant à en trouver des solutions prend la forme suivante :

1. Une population initiale S_0 de solutions, pas particulièrement optimales, est établie d'une manière peu coûteuse en ressources (p. ex. au hasard, à tâtons, par une autre méthode). Chacune de ces solutions prend normalement la forme d'une chaîne de caractères, par exemple une chaîne de bits.
2. Une fonction calcule pour chaque solution de l'étape précédente sa valeur adaptative (*fitness*), en relation directe avec l'optimalité de la solution. *Grosso modo*, plus une solution est près de la solution idéale, plus sa valeur adaptative sera élevée.
3. Au hasard, on choisit des couples de solutions que l'on combine afin d'obtenir une nouvelle solution. Notons d'une part qu'une solution ayant une valeur adaptative élevée aura plus de chance d'être choisie ; et d'autre part, que la nouvelle solution est construite en combinant une moitié de chacune des deux solutions « parents ».
4. Des segments de nouvelles solutions sont interchangeables, afin de simuler le phénomène génétique d'enjambement (*crossing-over*) ; de même, certains caractères d'une solution peuvent changer de manière aléatoire (p. ex. un « 0 » se transforme en « 1 »), une forme de mutation.
5. Ayant maintenant une population dont l'optimalité risque d'être plus élevée, on passe à l'itération suivante.

Dans le cas qui nous intéresse, des tournois successifs du dilemme itéré du prisonnier confrontent des stratégies obtenues par combinaison « génétique » des stratégies gagnantes du tournoi précédent. Ce qui est intéressant pour notre propos n'est pas le résultat d'une telle approche – mais notons qu'une stratégie très proche du TIT FOR TAT gagnant émerge – mais plutôt qu'il s'agit d'une relaxation d'une idéalisation, élément de l'ensemble de correction (Humphreys, 2004, p. 78). Mais ne s'agit-il que de cela ? Est-il possible que cette amélioration touche non seulement les idéalizations, mais également l'ontologie et donc, qu'elle engendre un nouveau modèle ? Laissons cette question en suspens, pour y revenir à la prochaine section (2.2.4).

⁷Le lecteur désirant plus de détails est invité à consulter un bon manuel d'intelligence artificielle, p. ex. (Russell et Norvig, 2003, p. 116-9).

Un dernier exemple, encore de relaxation d'une idéalisation, consiste en une expérience ultérieure (Wu et Axelrod, 1995), dans laquelle du bruit est introduit dans les interactions entre joueurs. Ainsi, à chaque interaction, un joueur, avec une faible probabilité (p. ex. 1%), peut voir l'action inverse de celle choisie être exécutée (un joueur voulant faire défection se voit coopérer malgré lui, ou vice-versa).

À la lumière de ces trois exemples, une conclusion s'impose : le concept d'ensemble de correction doit être légèrement élargi. Rappelons que

« [t]he principal purpose of the correction set is to indicate, in advance, the ways in which the computational template will need to be adjusted when (not if) it fails to match the empirical data. » (Humphreys, 2004, p. 78)

Or, les trois raffinements présentés dans cette section ne visaient pas – à notre connaissance – à améliorer l'adéquation du patron computationnel à des données empiriques *existantes*, mais seulement *potentielles*. Prenons le dernier raffinement, l'ajout de bruit. Cette amélioration ne s'est pas imposée pour répondre au défi de données réelles, c'est-à-dire actuellement disponibles. Plutôt, à la lumière d'exemples historiques⁸, il semble important – en fait, « *a vital research question in game theory* » – de modéliser adéquatement le bruit (Wu et Axelrod, 1995). Il s'agit donc de modifier un patron computationnel afin d'améliorer son adéquation à des données empiriques, non pas actuellement, mais plutôt potentiellement ou même probablement disponibles. La définition de l'ensemble de correction devrait donc refléter cette nuance.

Un autre aspect de cette définition doit demeurer, pour notre part, insoluble : l'idée que l'ensemble de correction est établi d'avance (« *in advance* », dans la définition citée au-dessus), lors de la construction du patron. Le fait qu'Axelrod (1984, 1997) n'explicite pas cette lucidité *a priori* ne saurait constituer un argument contre l'existence de celle-ci. Mais établir l'apriorité de l'ensemble de correction n'est pas, à proprement parler, une question de philosophie des sciences, mais plutôt d'histoire – et pourquoi pas, de sociologie – des sciences. La définition de l'ensemble de correction devrait donc, selon nous, se montrer plus prudente à cet égard et stipuler que l'ensemble de correction, s'il n'a pas été établi en même temps que la construction du patron, aurait au moins pu l'être. Cette apriorité en principe plutôt qu'en pratique respecte selon nous l'esprit de la conception originale de Humphreys (2004), tout en se montrant plus inclusive face à des cas comme celui qui nous occupe.

2.2.4 Patron ou patrons ?

À la section précédente (2.2.3), nous avons examiné différentes modifications dont une, l'utilisation d'algorithmes génétiques pour générer des stratégies de plus en plus performantes. Cette approche évolutionnaire (Axelrod, 1987) ainsi que l'approche écologique (Axelrod, 1984,

⁸Wu et Axelrod (1995) donne l'exemple de la destruction en 1983, par l'aviation militaire soviétique, d'un avion commercial sud-coréen égaré dans l'espace aérien soviétique. La décision de pénétrer dans un espace aérien contrôlé était une erreur – c'est-à-dire le fruit du bruit. Ainsi, en présence de bruit, il est difficile de savoir si l'adversaire a véritablement choisi la décision réalisée (p. ex. faire défection) ou s'il s'agit d'une erreur dû au bruit – ce qui est plus pardonnable.

p. 49-53) seront examinées dans cette section, afin de déterminer si elles constituent plusieurs patrons computationnels – ou, au contraire, s’il ne s’agit que de raffinements successifs d’un même patron.

Examinons en premier lieu l’approche écologique (Axelrod, 1984, p. 49-53). Cette approche consiste à simuler plusieurs tours successifs d’un tournoi ; les participants d’un tour donné sont choisis parmi les meilleurs participants du tour précédent, au *pro rata* du score obtenu à ce dernier tour. L’intention de base d’une telle approche est de déterminer si une stratégie (utilisée par des participants) deviendra dominante à l’issue du tournoi.

À la lumière de ces précisions, l’utilisation de la métaphore écologique prend tout son sens. Il s’agit, *grosso modo*, de la simulation (sous la forme d’un tournoi) d’un écosystème comprenant des individus (participants) appartenant à différentes espèces (stratégies). Les individus membres des espèces les plus adaptées (obtenant le plus haut score) se reproduiront plus : ces espèces gagnantes seront alors représentées en proportion plus grande au tour suivant.

Notons que seule la fréquence des stratégies varie d’un tour à un autre. En particulier, de nouvelles stratégies n’apparaissent pas spontanément à une génération. Le seul changement intergénérationnel possible est la disparition d’une stratégie (dont la proportion a chuté à zéro).

Nous devons maintenant déterminer si cette approche requiert un changement dans l’ontologie. La question est d’intérêt, car

« [c]hanges in the ontology will often require changes in the other elements of the correction set and come very close to starting a new construction process. »
(Humphreys, 2004, p. 79)

Ainsi, si nous concluons qu’il y a bel et bien eu un changement dans l’ontologie, nous serons en mesure d’affirmer qu’il y a probablement plusieurs patrons computationnels du dilemme itéré du prisonnier utilisés dans notre cas d’étude, les expériences d’Axelrod.

Rappelons que l’ontologie initiale (section 2.2.2), celle de la première expérience d’Axelrod (1980a), ne comptait que deux types d’entités : d’une part, des agents, ayant à cœur leur intérêt, pouvant se reconnaître entre eux ou du moins limiter leurs interactions à certains autres agents, et pouvant d’une manière ou d’une autre se « souvenir » des interactions passées ; d’autre part, des interactions entre agents.

Est-ce que l’approche écologique enrichit cette ontologie d’autres entités ou relations entre entités ? Nous pouvons identifier deux éléments nouveaux. D’une part, la génération, c’est-à-dire l’itération au terme duquel ont lieu les événements de « reproduction », semble une relation centrale à l’approche écologique. En effet, la présence d’un agent à une génération donnée ne peut être expliquée que par sa victoire (relative) à la génération précédente.

D’autre part, nous devons considérer l’interprétation qui est faite du score obtenu. Alors que dans les expériences antérieures (Axelrod, 1980a,b), le score n’est pas le sujet d’une interprétation particulière et ne sert *in fine* qu’à déterminer la stratégie « gagnante », il en est tout autrement dans l’approche écologique. En effet, étant donné que le score obtenu par une stratégie à une génération donnée détermine la proportion des agents qui utiliseront cette stratégie à

la génération suivante, une conclusion s'impose : le score n'est ni plus ni moins que la VALEUR ADAPTATIVE⁹ (*fitness*) de la stratégie !

Mais est-ce qu'un changement dans l'interprétation d'un patron constitue le point de départ d'un nouveau patron ou d'un nouveau modèle ? D'un côté, Humphreys (2004, p. 103) place l'interprétation parmi les six éléments de sa formalisation du modèle computationnel (vue à la section 1.2.4). On pourrait donc penser qu'un changement d'interprétation, s'il peut engendrer un nouveau modèle, laisse cependant le patron sous-jacent intact.

D'un autre côté, rappelons que l'interprétation joue un rôle important dans la justification d'un patron computationnel :

« To adopt the detachable interpretation view is to deeply misrepresent the epistemic status that templates have in scientific work. [...] To peel off the intended interpretation is to remove the justification we have for adopting the equations. »
(Humphreys, 2004, p. 80)

L'interprétation semble donc faire partie intégrante du patron ; si on change la première, on changerait le second. Mais un doute subsiste, qui ne s'effacera que lorsque Humphreys clarifiera la relation entre l'interprétation et l'émergence de nouveaux patrons. Notre propos, heureusement, ne nécessite pas une telle explicitation : en effet, puisque nous avons déjà établi que l'ontologie était enrichie par la relation de génération, nous pouvons conclure que l'approche écologique est probablement fondée sur un nouveau patron computationnel.

Avant de nous pencher sur les conséquences d'un tel dédoublement de patrons, examinons un second exemple : l'approche évolutionnaire. Cette approche ressemble à l'approche écologique, mais avec un élément supplémentaire (tiré de l'algorithmique génétique, présentée à la section 2.2.3) : entre deux tours successifs du tournoi, de nouvelles stratégies peuvent apparaître, par combinaison des stratégies gagnantes. Ainsi, à l'opposé de l'approche écologique, les changements intergénérationnels ne se limitent pas à de simples variations dans la fréquence des différentes stratégies.

Rappelons que nous avons déjà établi (à la section 2.2.3) que cette approche évolutionnaire constitue une relaxation d'une idéalisation et donc, d'une application de l'ensemble de correction. D'emblée, il est clair que l'approche évolutionnaire intègre l'approche écologique et donc, qu'elle aussi a une ontologie différente de celle des expériences initiales d'Axelrod (1980a,b). En effet, l'approche évolutionnaire partage avec l'approche écologique la relation de génération, ainsi que l'interprétation du score *qua* valeur évolutive. Deux autres éléments s'ajoutent également à l'ontologie de l'approche évolutionnaire :

- Alors qu'auparavant, on pouvait identifier l'agent et sa stratégie, l'approche évolutionnaire nous oblige à considérer la stratégie comme étant une entité à part entière, susceptible d'être propagée aux générations futures. L'approche évolutionnaire nous force en

⁹Plusieurs débats s'attaquent à la définition de la valeur adaptative ; le lecteur intéressé pourra consulter (Rosenberg et Bouchard, 2002) pour un survol complet et récent. Dans le cadre de ce mémoire, nous utilisons le concept de valeur adaptative au sens de succès reproductif réalisé. Bien qu'il ne s'agisse pas de la seule interprétation possible, elle a le mérite d'être celle utilisée par Axelrod (1984, 1997).

quelque sorte à nous désintéresser de l'individu pour nous concentrer sur le gène (Dawkins, 2006). L'ontologie est donc enrichie d'une nouvelle entité : la stratégie, en tant que chaîne de symboles (information représentant le « programme » que suit ou exécute l'agent) susceptible d'être transmise, après modifications et altérations.

- Ces mêmes modifications et altérations constituent des relations qui s'ajoutent aussi à l'ontologie : mutation et enjambement (*crossing-over*) sont des éléments de l'ontologie de l'algorithmique génétique qui migre vers celle de l'approche évolutionnaire.

L'approche évolutionnaire est donc elle aussi probablement fondée sur un nouveau patron, à la fois différent du patron original des premières expériences d'Axelrod (1980a,b), mais aussi différent du patron de l'approche écologique. À la prochaine section, nous aborderons la question des relations entre patrons apparentés.

2.2.5 Relation entre patrons apparentés

La section précédente (2.2.4) nous a permis de découvrir qu'il semble y avoir non pas un mais bien trois patrons computationnels mis en oeuvre dans les expériences d'Axelrod (1984, 1997). Dans cette section, nous nous intéresserons aux relations entre ces patrons apparentés. Nous présenterons trois perspectives alternatives sur la question.

En premier lieu, on peut considérer que les trois patrons computationnels partagent le même patron théorique. Rappelons (section 1.2.1) que le patron théorique est une relation abstraite et générale entre entités, qu'un patron computationnel correspondant rend applicable et tractable (Humphreys, 2004, p. 60-1). En particulier,

« [t]he most important kind of computational template is found at the first level at which a tractable mathematical form occurs as a result of substitutions into a theoretical template. Above that level, there can be great representational power but no direct application. » (Humphreys, 2004, p. 61)

Dans cette optique, le patron computationnel du dilemme itéré du prisonnier, ou patron de base, tel qu'utilisé dans les premières expériences d'Axelrod (1980a,b), serait un tel patron computationnel à la limite du théorique. Cela expliquerait d'une part la simplicité de son ontologie (qui ne compte que deux types d'éléments), et d'autre part la relation avec les deux autres patrons computationnels : autant ce patron de base que le patron computationnel de l'approche écologique et que celui de l'approche évolutionnaire seraient issus du même patron théorique.

Malheureusement, une telle perspective ne résiste pas à la critique. Car la différence entre les trois patrons n'en est pas seulement une d'applicabilité ou de tractabilité : il y a des différences au niveau de l'ontologie, qui laissent présager que la distinction entre ces patrons doit être beaucoup plus marquée.

Une deuxième approche est de voir dans la progression des patrons une relation de parenté prenant la forme d'une arborescence. Ainsi, le patron théorique de base donne naissance, non seulement à un patron computationnel correspondant qui sera utilisé expérimentalement, mais aussi à un nouveau patron théorique, celui de l'approche écologique. Ce nouveau patron est ca-

ractérisé par une ontologie enrichie de la relation de génération, ainsi que d'une interprétation du score obtenu en terme de valeur adaptative. De même, le patron théorique de l'approche écologique donne naissance à son tour au patron théorique de l'approche évolutionnaire, en ajoutant à l'ontologie les entités de stratégie et d'opérations génétiques (de mutation et d'enjambement).

Une telle perspective a le mérite d'explicitier complètement les relations entre patrons. Seule ombre au tableau : Humphreys (2004) reste muet sur la question des relations entre patrons théoriques¹⁰. Mais plutôt que d'être une critique de la perspective présentée ici, ce silence trahit plutôt une lacune dans la conception des modèles computationnels, qui devrait être comblée.

Allons plus loin : une troisième perspective accepte non seulement qu'un patron théorique en engendre un autre, mais en plus, qu'un patron théorique soit issu de plus d'un patron théorique ! Par exemple, le patron théorique de l'approche évolutionnaire serait issu, d'une part du patron théorique de l'approche écologique (qui fournit le cadre du dilemme itéré du prisonnier, avec reproduction accrue des meilleures stratégies) et d'autre part du patron théorique de l'algorithme génétique (qui fournit le cadre permettant la représentation symbolique des stratégies, avec reproduction par mutation et enjambement).

Des travaux récents dans le domaine de la simulation en nanoscience (Winsberg, 2006) semblent abonder dans ce sens. En raison de contraintes computationnelles, un phénomène de l'ordre du micron doit être simulé à plusieurs échelles différentes (*multiscale modeling*) : p. ex. au niveau des solides macroscopiques, au niveau moléculaire et au niveau atomique. Mais à chacun de ces niveaux correspond un patron théorique différent (p. ex. la mécanique classique, la mécanique quantique, etc.). Pire, ces patrons peuvent être mutuellement incohérents (mais un *handshaking algorithm* permet de créer un gradient à la frontière entre deux niveaux, où un patron cède progressivement sa place à un autre). Donc, nous avons ici un autre exemple de simulation informatique – et donc, par définition, de modèle computationnel – basée sur plus d'un patron théorique.

Une conclusion s'impose : s'il y a plusieurs patrons en oeuvre – et cela semble probable – nous sommes alors bel et bien en présence de trois patrons théoriques, et non d'un seul patron théorique avec trois patrons computationnels correspondants. La conception de Humphreys (2004) ne permet pas de trancher sur la nature exacte des relations entre ces trois patrons théoriques. Cependant, la présente section, et en particulier les deux dernières perspectives, suggère quelques pistes à explorer.

2.2.6 Le modèle du dilemme itéré du prisonnier

Il est temps de rassembler tous les éléments dégagés dans les sections précédentes en un tout cohérent : le modèle computationnel du dilemme itéré du prisonnier.

Rappelons qu'un modèle computationnel comprend six éléments : un patron computationnel ; des suppositions de construction ; un ensemble de correction ; une interprétation ; une jus-

¹⁰Tout au plus se contente-t-il de préciser que lors de la révision d'un patron, une modification de l'ontologie de celui-ci « *come very close to starting a new construction process* » (Humphreys, 2004, p. 79), sans expliciter la relation entre le patron original et le nouveau patron.

tification initiale ; et une représentation de sortie (Humphreys, 2004, p. 103). Les trois premiers éléments ayant déjà été explicités (sections 2.2.1, 2.2.2 et 2.2.3), nous nous attarderons dans cette section aux trois autres éléments.

En premier lieu, l'interprétation. L'ontologie (élément des suppositions de construction) nous invite à interpréter les entités mises en oeuvre comme étant des agents ayant à coeur leur intérêt, et interagissant deux à deux de manière répétée. Certaines variantes du patron – les approches écologiques (Axelrod, 1984, p. 49-53) et évolutionnaires (Axelrod, 1987) – requièrent que l'on interprète le score obtenu en tant que valeur adaptative (*fitness*).

D'où notre deuxième préoccupation, la justification initiale, qui découle tout naturellement de cette interprétation. En effet, si on choisit d'ignorer l'interprétation, il ne reste plus dans le modèle que des entités mathématiques interagissant selon des règles fixées par le patron théorique : il n'est alors plus possible de faire sens des suppositions de construction. Par exemple, si on ignore que le dilemme itéré du prisonnier se déroule entre couples de joueurs, on ne peut comprendre pourquoi la fonction $f_A(h_{A,k}, h_{B,k})$ est binaire plutôt que unaire, tertiaire, etc. Ce n'est qu'à la lumière de l'interprétation que l'on peut comprendre pourquoi cette fonction doit prendre en paramètre l'historique des interactions des deux joueurs.

Enfin, le dernier élément est la représentation de sortie, sur laquelle Axelrod (1984, 1997) reste muet. Peu nous importe la forme qu'elle prend, tant qu'elle existe – ce qui est le cas, sinon Axelrod n'aurait tout simplement pas pu rendre compte des résultats de ses expériences ! Ayant maintenant rassemblé différents éléments en un modèle computationnel cohérent, nous pouvons maintenant nous attarder sur le statut de simulation informatique qu'ont les expériences d'Axelrod.

2.3 La question de la simulation

Notre objectif jusqu'ici était de montrer dans quelle mesure la conception de modèle computationnel de Humphreys (2004) s'applique aux expériences d'Axelrod (1984, 1997). Cet objectif a été atteint en montrant en quoi le dilemme itéré du prisonnier constitue un modèle computationnel. Mais les expériences d'Axelrod sont plus qu'un simple modèle computationnel : elles sont aussi des simulations informatiques. Dans la présente section, nous nous pencherons donc sur ce statut particulier.

En premier lieu, les expériences d'Axelrod constituent-elles des simulations informatiques au sens de Humphreys ? Reportons-nous à la définition (présentée à la section 1.5.1) que propose ce dernier : un dispositif concret de calcul S produisant, par un processus temporel dynamique, des solutions à un modèle computationnel M représentant lui-même de manière statique ou dynamique un objet ou processus B , est une simulation (basique) (Humphreys, 2004, p. 110).

En décortiquant cette définition, on constate que les expériences d'Axelrod constituent une simulation informatique. D'une part, ces expériences, en tant que programmes informatiques devant nécessairement être exécutés sur un ordinateur, constituent des dispositifs concrets de

calcul. D'ailleurs, une preuve supplémentaire est le rôle joué par les contraintes computationnelles. Rappelons que

« [w]hen originally run in 1978, the round-robin tournament pushed the capabilities of a large mainframe computer. In 1994, the equivalent amount of computation was an easy job on a personal computer. » (Axelrod, 1997, p. 32)

Une telle amélioration des performances, en montrant le rôle des contraintes technologiques, vient renforcer le caractère concret du dispositif de calcul.

D'autre part, les expériences d'Axelrod, en tant que tournois entre agents interagissant itérativement, sont clairement des processus dynamiques et temporels. Ceci est encore plus visible dans les approches écologiques (Axelrod, 1984, p. 49-53) et évolutionnaires (Axelrod, 1987) que dans l'approche de base (Axelrod, 1980a) : un processus faisant intervenir des générations d'agents, avec « survie » des meilleurs, est manifestement dynamique et temporel. Autre argument en faveur de la nature dynamique et temporelle : les fonctions représentant les stratégies des joueurs peuvent être définies par récurrence (section 2.2.1).

Nous avons déjà présenté (à la section 2.2.6) le modèle computationnel du dilemme itéré du prisonnier. Or, les expériences d'Axelrod mènent celui-ci à identifier plusieurs propriétés (p. ex. la gentillesse, le pardon) que partagent les stratégies les plus performantes. Il semble clair que ces expériences permettent donc de trouver des stratégies gagnantes, bref des solutions, au dilemme itéré du prisonnier. Ainsi, les expériences d'Axelrod génèrent des solutions à un modèle computationnel. Notons que, puisque les stratégies utilisées lors des expériences d'Axelrod ne représentent – finitude oblige – qu'une infime proportion des stratégies possibles, on ne peut donc affirmer que toutes les solutions optimales ont été identifiées. En d'autres termes, ces expériences permettent d'identifier des conditions suffisantes – mais pas des conditions nécessaires – de l'émergence de la coopération. Ceci confirme la mise en garde de Humphreys :

« In fact, [...] because the goal of many agent-based procedures is to find a set of conditions that is sufficient to reproduce the behavior, rather than to isolate conditions which are necessary to achieve the result, a misplaced sense of understanding is always a danger. » (Humphreys, 2004, p. 132)

Les conditions suffisantes identifiées par les expériences peuvent aussi être démontrées nécessaires – mais par des approches analytiques et non par des simulations. D'ailleurs, un volet des travaux d'Axelrod consiste justement en l'analyse mathématique des propriétés identifiées d'abord dans le cadre des simulations¹¹.

Enfin, le modèle computationnel du dilemme itéré du prisonnier – est-ce utile de le rappeler ? – vise à représenter un processus réel, celui d'agents interagissant deux à deux et de manière répétée ; ce modèle remplit donc la quatrième condition de la définition.

Ayant satisfait aux quatre clauses de la définition, nous concluons que les expériences d'Axelrod constituent bel et bien des simulations informatiques au sens de Humphreys (2004). À la lumière de ce constat, une deuxième question se pose : dans quelle mesure les expériences

¹¹L'annexe B de (Axelrod, 1984) présente les propositions théoriques ainsi que leur démonstration respective.

d'Axelrod peuvent-elles nous aider à comprendre les simulations informatiques en tant qu'instruments ? Plus précisément, notre discussion sur la question (section 1.5.2), qui établissait de manière générale cette caractérisation, s'applique-t-elle au cas particulier des expériences d'Axelrod ?

D'une part, examinons l'imprégnation théorique sous-jacente à ces expériences. Nous avons déjà souligné (section 2.2.2) que les suppositions de construction du patron du dilemme itéré du prisonnier ne comptent ni lois, ni principes de base. Ce patron est donc dénué de toute dimension nomologique et donc, à cet égard, de toute imprégnation théorique.

Cependant, la simulation qui est liée à ce patron – que ce soit l'approche basique (Axelrod, 1980a), écologique (Axelrod, 1984, p. 49-53) ou évolutionnaire (Axelrod, 1987) – demeure entièrement déterminée par des constructions artificielles. Aucune donnée empirique n'entre en ligne de compte ; aucune interaction avec un système réel ne se produit ; aucune boucle de rétroaction entre simulation et réalité ne s'instaure. Bref, les simulations d'Axelrod se déroulent en vase clos, exemples privilégiés de ce que Humphreys qualifie d'études exploratoires (Humphreys, 2004, p. 116).

Sous ces conditions, on peut parler d'une imprégnation théorique – par opposition à empirique – qui baigne toute utilisation de ces instruments que sont les simulations d'Axelrod. Ainsi, *contra* Hacking et en accord avec nos conclusions générales sur la question (section 1.5.2), il faut connaître le fonctionnement de l'instrument pour l'utiliser, puisque cet instrument, en fait, est entièrement réductible à ses principes de fonctionnement : l'instrument n'est qu'un ensemble de règles, sans aucune donnée empirique !

Ce qui nous emmène naturellement au deuxième et dernier point que nous voulons aborder : la question de la justification d'une simulation *qua* instrument. Nous avons déjà établi (section 1.5.2) que, de manière générale, une telle justification s'effectue selon la construction (Humphreys), l'utilisation (Hacking) ainsi que la réputation (Hacking, Winsberg). Les simulations d'Axelrod respectent-elles ce schéma ?

D'une part, Axelrod souligne l'omniprésence du dilemme itéré du prisonnier en psychologie : « [t]he iterated Prisoner's Dilemma has become the *E. coli* of social psychology. » (Axelrod, 1984, p. 28) La réputation de ce « *standard paradigm for studying issues in fields as diverse as evolutionary biology and networked computer systems* » (Axelrod, 1997, p. xi) n'est donc plus à faire. Le rôle de cette réputation dans le déroulement de l'entreprise scientifique demeure, nous l'avons souligné (section 1.5.2), une question d'histoire et de sociologie des sciences. Bornons-nous simplement à mentionner cette réputation et l'importance que semble y apporter Axelrod lui-même.

De même pour la contribution de l'utilisation de l'instrument à la justification de celui-ci. On sait qu'Axelrod a effectué des expériences répétées sur de longues périodes de temps : entre les premières (Axelrod, 1980a) et les plus récentes (Wu et Axelrod, 1995) expériences, quinze ans se sont écoulés. Il serait étonnant qu'une telle familiarité n'ait pas engendré une certaine confiance dans l'instrument ; comme l'écrit Hacking sur la microscopie,

« [p]ractice [...] creates the ability to distinguish between visible artefacts of the preparation or the instrument, and the real structure that is seen with the microscope. This practical ability breeds conviction. » (Hacking, 1981, p. 309)

Mais ultimement, la contribution de cette utilisation à la justification complète de la simulation demeure une question relevant plus de la psychologie et de la sociologie que de la philosophie des sciences. Que cette contribution soit non nulle, certes ; mais rien de plus ne peut en être dit.

Enfin, nous avons déjà établi (dans la présente section) que les expériences d'Axelrod se déroulent, en quelque sorte, en vase clos : aucun recours à des données empiriques n'est effectué. Cela réduit considérablement les sources possibles de justification : puisque l'instrument est en quelque sorte sa propre source de justification, la construction de cet instrument devient une composante importante de toute justification potentielle. Humphreys (2004) souligne la participation du processus de construction à la justification *initiale* ; mais dans le cas de simulations informatiques exploratoires, ce processus de construction peut contribuer de manière non négligeable à *toute* justification de l'instrument.

Résumons : la justification des expériences d'Axelrod proviendrait en grande partie de la construction du modèle computationnel sous-jacent, avec une contribution non nulle, peut-être même importante (c'est ce que semble dire Axelrod, avec sa remarque sur le « E. coli »), de l'utilisation répétée et de la réputation.

2.4 Le rôle épistémique des simulations

Au terme de cette analyse, il ne sera pas surprenant d'arriver à la conclusion que la simulation d'Axelrod joue un rôle épistémique relevant plutôt de l'intuition que de l'explication, et ce pour deux raisons.

D'une part, nous avons vu (à la section 2.3) que les expériences d'Axelrod, en tant que telles (c'est-à-dire sans l'analyse théorique qui en a été tirée), ne peuvent identifier des conditions nécessaires d'émergence de la coopération : seules des conditions suffisantes peuvent être dégagées des résultats. Ainsi, l'apport épistémique de ces expériences, eu égard à des phénomènes de la biologie évolutionnaire, risque de tenir plus des « *just-so-stories* » pan-adaptationnistes (Dennett, 1995, p. 242) que de l'explication causale. Ces expériences ne peuvent nous dire comment la coopération a émergé, mais seulement comment il est possible qu'elle ait émergé.

Cet apport intuitif plutôt qu'explicatif est encore plus visible dans l'application des résultats aux systèmes biologiques. Axelrod et Hamilton (1981) s'intéressent aux conditions d'émergence – aux raisons d'être – de phénomènes aussi divers que le mutualisme, la territorialité, la reconnaissance du visage et les causes du cancer. Dans chaque cas, les deux auteurs présentent un lien possible avec le dilemme itéré du prisonnier. Mais de leur propre aveu, ce lien demeure un lien suffisant et non nécessaire : le dilemme itéré du prisonnier est une intuition, une narration, de l'émergence possible de ces différents phénomènes.

À cet égard, l'apport de Maynard Smith est intéressant. Son analyse (1982, p. 167-73) des premières expériences d'Axelrod (1981 ; Axelrod et Hamilton, 1981), appliquée au cas des babouins et des humains, souligne le fait que non pas un mais trois mécanismes de transmission des stratégies sont possibles : génétique, par apprentissage et culturel. Cette sous-détermination fait en sorte qu'il n'est même pas clair que les expériences d'Axelrod portent sur la biologie évolutionnaire (plutôt que sur la psychologie de l'apprentissage ou sur l'anthropologie, p. ex.). Ainsi, pour que ces expériences puissent résulter en une explication, il faudrait tout d'abord identifier le domaine sur lequel porte l'explication. Jusqu'à ce que cela soit fait, les résultats de ces expériences ne peuvent constituer qu'une aide à la réflexion.

D'autre part, nos conclusions sur les sources de justification des simulations d'Axelrod viennent renforcer l'idée que ces simulations agissent comme « pompe à intuition¹² » plutôt que comme explication. En effet, ces simulations sont principalement justifiées par le processus de construction du patron théorique sous-jacent (section 2.3) ; également, ce même patron théorique n'est fondé ni sur des lois, ni sur des principes tirés de données empiriques (section 2.2.2) ; donc, ces simulations, largement instrumentales, au mieux ne peuvent jouer qu'un rôle régulateur.

Ainsi, notre application de la conception de Humphreys (2004) aux diverses expériences d'Axelrod (1984, 1997) nous permet de conclure que plusieurs facteurs importants peuvent cantonner la simulation informatique dans un rôle exploratoire plutôt qu'explicatif. Bien que nous n'ayons considéré qu'une simulation particulière appliquée à une seule théorie (celle de l'évolution), il demeure que ces facteurs identifiés sont tout sauf exceptionnels. Nous avançons donc l'idée que les simulations informatiques au sens de Humphreys (2004), en biologie évolutionnaire comme dans d'autres domaines, risquent de jouer un rôle épistémique surtout instrumental.

2.5 Conclusion

Ce chapitre consacré à l'application de la conception de Humphreys aux diverses expériences d'Axelrod nous a permis d'atteindre notre triple objectif.

D'une part, après avoir présenté le dilemme du prisonnier ainsi que les simulations d'Axelrod (1984, 1997), nous avons pu dégager un modèle computationnel de dilemme du prisonnier. Ceci nous a mené à montrer que les travaux d'Axelrod (1984, 1997) constituent des simulations informatiques au sens de Humphreys (2004). Cette démonstration nous a permis d'établir que la conception de l'empirisme computationnel de Humphreys (2004), que ce dernier avait illustré surtout avec des exemples tirés de la physique, est assez générale pour englober au moins une théorie (la théorie de l'évolution) d'un autre domaine (nommément la biologie évolutionnaire).

Cependant, et c'est notre deuxième point, notre application de la conception de Humphreys (2004) à un cas d'étude nous a permis de mettre en lumière certaines critiques mineures, concer-

¹²Nous traduisons ainsi l'expression « *intuition pump* », fréquemment utilisée par Dennett (voir par exemple (Dennett, 1995, p. 141)).

nant la définition de l'ensemble de correction, les relations entre patrons apparentés ainsi que la justification des études exploratoires telles que les simulations d'Axelrod (1984, 1997).

Enfin, cette mise en oeuvre de la conception de Humphreys (2004) nous a donné les outils permettant de la nuancer, mais aussi de l'utiliser pour élucider le rôle épistémique d'une simulation informatique portant sur la théorie de l'évolution. Au prochain chapitre, nous effectuerons un retour sur cette question de recherche qui nous a préoccupé tout au long de ce mémoire.

Discussion et conclusion

Arrivés au terme du parcours que nous nous étions fixé initialement, il est maintenant temps de tracer un portrait général des acquis effectués. Pour ce faire, nous suivrons un plan similaire au plan du mémoire lui-même : après avoir abordé les principales conclusions que l'on peut tirer de notre analyse de la conception de Humphreys (2004), nous en examinerons quelques conséquences et critiques. Puis, nous tournant vers les expériences d'Axelrod (1984, 1997), nous examinerons comment la conception de Humphreys (2004) permet d'éclairer le rôle épistémique des simulations informatiques dans le domaine biologique. L'accent sera mis sur les faiblesses de la conception de Humphreys (2004), identifiées à la lumière de cette illustration par l'exemple des travaux d'Axelrod (1984, 1997).

Sans doute l'un des résultats les plus intéressants de notre analyse de la conception de Humphreys (2004) constitue le dégagement d'une définition adéquate de la simulation informatique (section 1.5.1) : un instrument servant à explorer expérimentalement les propriétés d'un modèle computationnel.

Ceci nous permet de mettre en évidence deux propriétés intéressantes des simulations informatiques au sens de Humphreys (2004). D'une part et *contra* Hacking (1981), il y a souvent imprégnation théorique lors de l'utilisation d'une simulation informatique (section 1.5.2). D'autre part, la justification d'une simulation provient non seulement de sa construction, mais également de son utilisation (répétée) et de sa réputation (section 1.5.2).

Au coeur de toute simulation informatique au sens de Humphreys (2004) se trouve un modèle computationnel et donc un patron computationnel, véritables pièces maîtresses du projet de l'empirisme computationnel (section 1.2.5). Or, l'adoption – ou la réalisation – de ce projet comporte un certain nombre de conséquences pour l'entreprise scientifique dans son ensemble.

En premier lieu, la conception syntaxique des théories scientifiques, qui avait été en partie éclipsée par la conception sémantique (section 1.1.2), est appelée à effectuer un retour au centre des préoccupations des philosophes des sciences. Mais il s'agit d'une conception syntaxique modifiée, faisant appel non pas à la logique formelle mais plutôt au langage mathématique, voire computationnel.

En second lieu, malgré ce virage syntaxique, la dimension nomologique des patrons et modèles computationnels demeure tributaire d'une dimension encore plus importante selon Humphreys (2004) : la dimension computationnelle et donc la tractabilité (sections 1.2.2 et 1.4.1). Sans nier, à l'instar de Cartwright (1983), l'existence des lois scientifiques, Humphreys (2004)

en réduit considérablement l'importance. Nous avons d'ailleurs considéré deux exemples de patrons computationnels dénués de toute dimension nomologique : le patron du processus de Poisson (section 1.2.2) et le patron du dilemme itéré du prisonnier (section 2.2.2).

En troisième lieu, en raison de la sous-détermination inhérente aux patrons théoriques (section 1.3.1), il semble nécessaire d'abandonner toute position réaliste pure, au profit, non pas d'un instrumentalisme qui déplairait à plusieurs, mais d'un réalisme sélectif (sections 1.3.2 et 1.4.2), qui reconnaît le réalisme de certaines parties, et l'instrumentalisme des autres parties d'une même théorie scientifique.

Toutes ces conséquences, cependant, peuvent n'être que temporaire. En effet, jamais Humphreys (2004) ne fait la preuve que l'adoption de l'empirisme computationnel doit être permanente : les mêmes percées technologiques dont il vante l'importance pourraient fort bien permettre de trouver des solutions analytiques – donc, *non* computationnelles – aux problèmes présentement solutionnés par ordinateur (section 1.3.4). Cependant, cette critique ne fait que limiter la portée de la conception de Humphreys (2004), sans l'affaiblir pour autant.

Suite à cette analyse de la conception de Humphreys (2004), nous avons présenté en détail les différentes expériences d'Axelrod (1984, 1997) et avons tenté d'illustrer la conception de l'empirisme computationnel par cette simulation canonique du domaine de la biologie évolutionnaire.

Or, cette mise en application nous a permis de constater que les expériences d'Axelrod (1984, 1997) respectent bel et bien le schéma de modèle computationnel (section 2.2.6) et constituent une simulation informatique au sens de Humphreys (section 2.3). Ainsi, la définition de simulation informatique proposée par Humphreys (2004) et mise en évidence par notre analyse, semble rendre compte adéquatement du statut épistémique des simulations informatiques en science, en particulier en biologie évolutionnaire, hormis quelques modifications sur lesquelles nous nous pencherons maintenant.

En effet, plusieurs faiblesses de la conception originale de Humphreys (2004), mineures certes mais tout de même intéressantes, ont été mises en lumière lors de l'application de cette conception à notre cas d'étude.

En premier lieu, nous avons identifié un problème avec la définition originale de l'ensemble de correction. Contrairement à ce que propose Humphreys (2004), il semble que l'ensemble de correction vise à améliorer l'adéquation du patron à des données empiriques *potentielles*, plutôt qu'*existantes* (section 2.2.3). Nous avons en effet vu les différentes améliorations apportées au modèle du dilemme itéré du prisonnier, modifications nullement motivées par une confrontation du modèle avec l'empirie.

De même, alors que la définition de l'ensemble de correction requiert que celui-ci soit construit en même temps que le patron lui-même, notre cas d'étude nous indique qu'il est sans doute préférable de n'exiger qu'une apriorité en principe : tant que l'ensemble de correction aurait pu être construit en même temps que le patron, l'esprit de la conception de Humphreys (2004) semble respecté, sans exiger que l'on connaisse dans le détail l'histoire de vie d'un

modèle computationnel (section 2.2.3).

Deux autres faiblesses de la conception de Humphreys (2004) touchent les relations entre patrons. D'une part, Humphreys (2004) n'explicite jamais sous quelles conditions un changement d'interprétation est suffisamment important pour entraîner un changement de patron. Or, notre cas d'étude nous a placé devant deux tels changements d'interprétation qui intuitivement, semblent appeler de nouveaux patrons (section 2.2.4).

D'autre part, Humphreys (2004) reste muet sur la question des rapports entre patrons théoriques. En particulier, notre cas d'étude a souligné la question de savoir si un patron théorique peut être dérivé à partir de deux patrons théoriques différentes – en d'autres termes, avoir deux parents (section 2.2.5).

Un autre point est soulevé par le fait que les expériences d'Axelrod (1984, 1997) constituent ce que Humphreys (2004) qualifie d'*étude exploratoire* : en effet, ces simulations se déroulent en vase clos, sans boucle de rétroaction avec l'empirie. Or, dans le cas de telles simulations « closes », la question de la justification devient épineuse. Notre cas d'étude semble montrer que non seulement la justification *initiale*, mais aussi la justification *complète* du modèle doit alors provenir du processus de construction, de la réputation et de l'utilisation (section 2.3). Notre discussion sur ces trois derniers aspects est donc d'autant plus importante dans le cas de simulations en vase clos.

Ces divergences mineures entre la conception originale de Humphreys (2004) telle qu'analysée au premier chapitre du présent mémoire, et notre cas d'étude (Axelrod, 1984, 1997) nous permettent de nuancer le rôle épistémique des simulations informatiques. Dans le cas de simulations exploratoires (c.-à-d. en vase clos), ce rôle épistémique semble relever plus de l'intuition que de l'explication (section 2.4). Ce qui ne saurait surprendre personne : après tout, de telles simulations, isolées de l'empirie, ne peuvent identifier que des conditions suffisantes – mais pas des conditions nécessaires – de l'émergence du phénomène à l'étude. En d'autres termes, elles ne peuvent fournir de *caractérisation*.

D'autre part, il nous apparaît important de revenir sur un point soulevé par Maynard Smith (1982). La sous-détermination du modèle d'Axelrod (1984, 1997) est à ce point grande qu'on ne peut même pas déterminer avec certitude de quelle discipline scientifique (biologie évolutionnaire ? psychologie de l'apprentissage ? ethnologie ?) il se réclame (section 2.4). Cette sous-détermination, on l'a vu (section 1.3.1), ne constitue pas pour Humphreys (2004) un obstacle mais plutôt une opportunité. De même, cette sous-détermination semble appuyer l'idée qu'une réorganisation des sciences, sur la base du patron utilisé plutôt que de l'objet d'étude, est possible et peut-être même imminente (section 1.4.3).

Cette urgence souligne l'importance d'approfondir certains aspects identifiés dans le présent mémoire. D'une part, il serait intéressant d'illustrer la conception de Humphreys (2004) par un exemple de simulation tiré d'une autre discipline, afin de rendre la conception de Humphreys (2004) encore plus robuste. D'autre part, les philosophes des sciences devraient se pencher sur la possibilité d'une réorganisation « computationnelle » des sciences : il s'agirait certainement

d'un changement de paradigme – voire de méta-paradigme – aux proportions coperniciennes.

Bibliographie

- AXELROD, R. M. 1980a. Effective Choice in the Prisoner's Dilemma. *Journal of Conflict Resolution* 24, 1, 3–25.
- AXELROD, R. M. 1980b. More Effective Choice in the Prisoner's Dilemma. *Journal of Conflict Resolution* 24, 3, 379–403.
- AXELROD, R. M. 1981. The Emergence of Cooperation among Egoists. *American Political Science Review* 75, 2, 306–318.
- AXELROD, R. M. 1984. *The Evolution of Cooperation*. Basic Books, New York.
- AXELROD, R. M. 1987. The Evolution of Strategies in the Iterated Prisoner's Dilemma. Dans *Genetic Algorithms and Simulated Annealing*, L. Davis (éd.). Pitman, London.
- AXELROD, R. M. 1997. *The Complexity of Cooperation*. Princeton University Press, Princeton (N.J.).
- AXELROD, R. M. ET HAMILTON, W. D. 1981. The Evolution of Cooperation in Biological Systems. *Science* 211, 1390–1396.
- BEDAU, M. 2003. Downward Causation and the Autonomy of Weak Emergence. *Principia Revista Internacional de Epistemologica* 6, 5–50.
- BENTZ, H.-J. 1980. Das Buffon-Nadelproblem (1777). *Praxis der Mathematik* 22, 6, 167–171.
- CARTWRIGHT, N. 1983. *How the Laws of Physics Lie*. Oxford University Press, Oxford.
- CARTWRIGHT, N. 1999. *The dappled world : a study of the boundaries of science*. Cambridge University Press, Cambridge.
- DAWKINS, R. 2006. *The Selfish Gene*, éd. trentième anniversaire. Oxford University Press, Oxford.
- DENNETT, D. C. 1995. *Darwin's Dangerous Idea : Evolution and the Meanings of Life*. Simon & Schuster, New York.
- FEYERABEND, P. 1978. *Against Method : Outline of an anarchistic theory of knowledge*. Verso, London.

- FEYNMAN, R., LEIGHTON, R. ET SANDS, M. 1965. *The Feynman Lectures on Physics*. Vol. 2. Addison-Wesley, Reading (Mass.).
- HACKING, I. 1981. Do We See Through a Microscope ? *Pacific Philosophical Quarterly* 62, 305–322.
- HACKING, I. 1989. *Concevoir et expérimenter : thèmes introductifs à la philosophie des sciences expérimentales*. Épistémè Essais. Christian Bourgeois, Paris.
- HACKING, I. 1992. Do Thought Experiments have a Life of Their Own ? Comments on James Brown, Nancy Nersessian and David Gooding. *Proceedings of the Biennial Meeting of the Philosophy of Science Association (PSA) 1992*, 2, 302–308.
- HARVEY, D., GREGORY, J., HOFFERT, M. ET JAIN, A. 1997. An Introduction to Simple Climate Models used in the IPCC Second Assessment Report. Rapport technique, Intergovernmental Panel on Climate Change.
- HEMPEL, C. 2004. *Éléments d'épistémologie*. Armand Colin, Paris.
- HUMPHREYS, P. 1990. Computer Simulations. *Proceedings of the Biennial Meeting of the Philosophy of Science Association (PSA) 1990*, 2, 497–506.
- HUMPHREYS, P. 1994. Numerical Experimentation. Dans *Patrick Suppes : Scientific Philosopher*, P. Humphreys (éd.). Vol. 2. Kluwer, Dordrecht, 103–121.
- HUMPHREYS, P. 1995a. Computational empiricism. *Foundations of Science* 1, 1, 119–130.
- HUMPHREYS, P. 1995b. Computational science and scientific method. *Minds and Machines* 5, 4, 499–512.
- HUMPHREYS, P. 2000. Extending Ourselves. Dans *Science at Century's End : Philosophical Questions on the Progress and Limits of Science*. University of Pittsburgh Press, Pittsburgh, 13–32.
- HUMPHREYS, P. 2002. Computational Models. *Philosophy of Science* 69, S1–S11.
- HUMPHREYS, P. 2004. *Extending Ourselves : Computational Science, Empiricism, and Scientific Method*. Oxford University Press, Oxford.
- HUMPHREYS, P. 2006. Self-Assembling Systems. *Philosophy of Science* 73, 595–604.
- HUMPHREYS, P. 2007. Epistemolog del Siglo XXI. *Anthropos* 214, 65–70.
- KELLER, E. F. 2002. *Making Sense of Life : Explaining Biological Development with Models, Metaphors, and Machines*. Harvard University Press, Cambridge (Mass.).
- KELLER, E. F. 2005. The century beyond the gene. *Journal of Biosciences* 30, 1, 101–108.

- KITANO, H. 2002. Computational systems biology. *Nature* 420, 206–210.
- KITCHER, P. 1989. Explanatory Unification and the Causal Structure of the World. Dans *Scientific Explanation*, P. Humphreys et W. Salmon (éd.). University of Minnesota Press, Minneapolis.
- KLINE, M. 1971. *Mathematical Thought from Ancient to Modern Times*. Oxford University Press, New York.
- KUHN, T. 1996. *The Structure of Scientific Revolutions*, troisième éd. University of Chicago Press, Chicago.
- LAKATOS, I. 1970. Falsification and the Methodology of Scientific Research Programs. Dans *Criticism and the Growth of Knowledge*, I. Lakatos et A. Musgrave (éd.). Cambridge University Press, Cambridge.
- LENHARD, J. 2006a. Controlling Complex Phenomena with the Help of Opaque Models. Dans *Models and Simulations (Paris, 2006)*.
- LENHARD, J. 2006b. Surprised by a Nanowire : Simulation, Control, and Understanding. *Philosophy of Science* 73, 605–616.
- LENHARD, J. 2007. Computer Simulation : The Cooperation between Experimenting and Modeling. *Philosophy of Science* 74, 176–194.
- LLOYD, E. A. 1994. *The Structure and Confirmation of Evolutionary Theory*. Princeton University Press, Princeton (N.J.).
- MAYNARD SMITH, J. 1982. *Evolution and the Theory of Games*. Cambridge University Press, Cambridge.
- MERTON, R. K. 1973. *The sociology of science : theoretical and empirical investigations*. University of Chicago Press, Chicago.
- OLSON, M. J. 1965. *The Logic of Collective Action*. Harvard University Press, Cambridge (Mass.).
- POPPER, K. R. 1978. *La logique de la découverte scientifique*. Payot, Paris.
- ROSENBERG, A. ET BOUCHARD, F. 2002. Fitness. Dans *The Stanford Encyclopedia of Philosophy*, éd. hiver 2002, E. N. Zalta (éd.). Adresse URL : <http://plato.stanford.edu/archives/win2002/entries/fitness/>.
- ROSS, S. M. 1997. *Initiation aux probabilités*. Presses polytechniques et universitaires romandes, Lausanne.

-
- RUSSELL, S. J. ET NORVIG, P. 2003. *Artificial Intelligence : A Modern Approach*. Pearson Education, Upper Saddle River (N.J.).
- SIPSER, M. 1997. *Introduction to the Theory of Computation*. PWS, Boston.
- SUPPES, P. 1967. What is a scientific theory ? Dans *Philosophy of science today*, S. Morgenbesser (éd.). Basic Books, New York, 55–67.
- SUPPES, P. 2002. Set-Theoretic Structures in Science. Dans *Representation and Invariance of Scientific Structures*, P. Suppes (éd.). CSLI, Stanford (Calif.).
- VAN FRASSEN, B. 1972. A Formal Approach to the Philosophy of Science. Dans *Paradigms and Paradoxes*, R. Colodny (éd.). University of Pittsburgh Press, Pittsburgh, 303–366.
- WILKES, M. V. 1956. *Automatic Digital Computers*. John Wiley & Sons, New York.
- WINSBERG, E. 2003. Simulated Experiments : Methodology for a Virtual World. *Philosophy of Science* 70, 1, 105–125.
- WINSBERG, E. 2006. Handshaking Your Way to the Top : Simulation at the Nanoscale. *Philosophy of Science* 73, 582–594.
- WOLFRAM, S. 2002. *A New Kind of Science*. Wolfram Media, Champaign (Illinois).
- WU, J. ET AXELROD, R. M. 1995. How to Cope with Noise in the Iterated Prisoner's Dilemma. *Journal of Conflict Resolution* 39, 1, 183–189.